A Program to Compare Statistically The Generalized Von Bertalanffy Growth Curves and Their Parameters

R. K. Misra

Modelling and Statistics Group
Biological Sciences Branch
Department of Fisheries and Oceans
P. O. Box 550
Halifax, Nova Scotia, Canada
B3J 2S7

July 1987

Canadian Technical Report of Fisheries and Aquatic Sciences No. 1439

Fisheries and Oceans Pêches et Océans

Canada

88-02789/

Canadian Technical Report of Fisheries and Aquatic Sciences

Technical reports contain scientific and technical information that contributes to existing knowledge but which is not normally appropriate for primary literature. Technical reports are directed primarily toward a worldwide audience and have an international distribution. No restriction is placed on subject matter and the series reflects the broad interests and policies of the Department of Fisheries and Oceans, namely, fisheries and aquatic sciences.

Technical reports may be cited as full publications. The correct citation appears above the abstract of each report. Each report is abstracted in *Aquatic Sciences and Fisheries Abstracts* and indexed in the Department's annual index to scientific and technical publications.

Numbers 1 456 in this series were issued as Technical Reports of the Fisheries Research Board of Canada. Numbers 457 714 were issued as Department of the Environment, Fisheries and Marine Service, Research and Development Directorate Technical Reports. Numbers 715 924 were issued as Department of Fisheries and the Environment, Fisheries and Marine Service Technical Reports. The current series name was changed with report number 925.

Technical reports are produced regionally but are numbered nationally. Requests for individual reports will be filled by the issuing establishment listed on the front cover and title page. Out-of-stock reports will be supplied for a fee by commercial agents.

Rapport technique canadien des sciences halieutiques et aquatiques

Les rapports techniques contiennent des renseignements scientifiques et techniques qui constituent une contribution aux connaissances actuelles, mais qui ne sont pas normalement appropriés pour la publication dans un journal scientifique. Les rapports techniques sont destinés essentiellement à un public international et ils sont distribués à cet échelon. Il n'y a aucune restriction quant au sujet; de fait, la série reflète la vaste gamme des intérêts et des politiques du ministère des Pêches et des Océans, c'est-à-dire les sciences halieutiques et aquatiques.

Les rapports techniques peuvent être cités comme des publications complètes. Le titre exact paraît au-dessus du résumé de chaque rapport. Les rapports techniques sont résumés dans la revue *Résumés des sciences aquatiques et halieutiques*, et ils sont classés dans l'index annual des publications scientifiques et techniques du Ministère.

Les numéros 1 à 456 de cette série ont été publiés à titre de rapports techniques de l'Office des recherches sur les pêcheries du Canada. Les numéros 457 à 714 sont parus à titre de rapports techniques de la Direction générale de la recherche et du développement, Service des pêches et de la mer, ministère de l'Environnement. Les numéros 715 à 924 ont été publiés à titre de rapports techniques du Service des pêches et de la mer, ministère des Pêches et de l'Environnement. Le nom actuel de la série a été établi lors de la parution du numéro 925.

Les rapports techniques sont produits à l'échelon régional, mais numérotés à l'échelon national. Les demandes de rapports seront satisfaites par l'établissement auteur dont le nom figure sur la couverture et la page du titre. Les rapports épuisés seront fournis contre rétribution par des agents commerciaux.

Canadian Technical Report of
Fisheries and Aquatic Sciences No. 1439

July 1987

A PROGRAM TO COMPARE STATISTICALLY THE GENERALIZED VON BERTALANFFY GROWTH CURVES AND THEIR PARAMETERS

R. K. Misra

Modelling and Statistics Group
Biological Sciences Branch
Department of Fisheries and Oceans
P.O. Box 550, Halifax
N. S., Canada, B3J 2S7

© Minister of Supply and Services Canada 1987
Cat. No. Fs 97-6/1439E ISSN 0706-6457

CONTENTS

LIST OF TABLES v

LIST OF ILLUSTRATIONS vi

ABSTRACT / RÉSUMÉ vii

1. INTRODUCTION 1

2. SUMMARY OF THE METHODOLOGY 1

3. DATA FLOW 3

4. INTERNAL STRUCTURE 3

5. COMMON BLOCK DESCRIPTIONS 3

6. INPUT PARAMETERS AND DATA 3

6.1 User Controlled Parameters 3
6.1.1 Parameters Describing Samples 3
6.1.2 Parameters Used For Setting Initial Values of Y_{1INF} and RHO 3
6.1.3 Parameters Used To Control Maximum Number of Various Iterations in the Fitting Procedure 4
6.1.4 Parameters Affecting Precision of Solution 4
6.1.5 Values Used For Selecting D Value 4
6.1.6 Other Parameters 4
6.2 Input Parameter Summary 4

6.3 Raw Data Input 4

7. MODULE DESCRIPTIONS 5

1.0 GVONB Main Program 5
2.0 INIT Subroutine 5
3.0 PLOT Subroutine 5
4.0 MISRA9 Subroutine 5
5.0 MISRA1 Subroutine 5
5.1 MISRA5 Subroutine 6
5.2 TRY1 Subroutine 6
5.3 TRY2 Subroutine 6
5.4 TRY3 Subroutine 6
5.5 MISRA6 Subroutine 6
5.6 MISRA7 Subroutine 6
5.7 MISRA2 Subroutine 6
5.8 Function Subprograms 6
5.9 SSP Subroutines 6
6.0 MISRA3 Subroutine 6
7.0 MISRA4 Subroutine 7

8. HOW TO RUN GVONB 7

9. EXPECTED OUTPUT 7

10. EXAMPLE RUN 7

11. PROGRAM LISTING 8

12. ACKNOWLEDGEMENTS 8

13. REFERENCES 8

14. APPENDIX 1 14

15. APPENDIX 2 16

16. APPENDIX 3 31

LIST OF TABLES

TABLE 1. Common blocks 9

TABLE 2. Input parameter summary 9

LIST OF ILLUSTRATIONS

FIG. 1. Data flow diagram for GVONB program 10

FIG. 2. Internal structure of GVONB program 10

FIG. 3. Internal structure of MISRA1 subroutine 11

FIG. 4. Internal structure of MISRA7 subroutine 12

FIG. 5. Internal structure of MISRA2 subroutine 12

FIG. 6. Internal structure of function subprograms 13

FIG. 7. Internal structure of SSP subroutines 13

ABSTRACT

Misra, R. K. 1987. A program to compare statistically the generalized von Bertalanffy growth curves and their parameters. Can. Tech. Rep. Fish. Aquat. Sci. No. 1439. vii + 89 p.

The von Bertalanffy growth formula (VBGF) is widely used within and outside the field of fishery science. Large flexibility built into the generalized VBGF enables it to provide a biologically realistic fit to data. A computer program was written to fit the generalized VBGF to a sample, to examine the fit statistically, and to compare statistically the curves fitted to several samples. Later, Wycove Systems Limited internally documented the program code and modified it to conform more closely to structured programming methods. A detailed account of the program and access to it, with an example, is presented here.

Key words: Computer program; generalized von Bertalanffy growth curves; fit; and comparison.

RÉSUMÉ

Misra, R. K. 1987. A program to compare statistically the generalized von Bertalanffy growth curves and their parameters. Can. Tech. Rep. Fish. Aquat. Sci. No. 1439. vii + 89 p.

La formule de croissance deon Bertalanffy (FCVB) est couramment utilisée dans le domaine des sciences halieutiques et à l'extérieur de ce domaine. La grande souplesse intrinsèque de la FCVB généralisée permet un lissage biologiquement réaliste des données. Un infoprogramme a été écrit pour ajuster la FCVB généralisée à un échantillon, pour examiner statistiquement le lissage et pour comparer statistiquement les courbes ajustées à plusieurs échantillons. Ultérieurement, la Wycove Systems Limited a établi une documentation interne et a modifié le programme de façon qu'il soit mieux adapté aux méthodes de programmation structurée. Une description détaillée du programme et des modes d'accès est présentée, avec un exemple.

1. INTRODUCTION

The study of growth is of fundamental importance in theoretical and applied ecology and is central to fishery biology. The von Bertalanffy growth formula, VBGF (von Bertalanffy 1938, 1960), plays a central role in studies of growth, in particular as a model for animal growth. It has been used in the general theory of fishing and has many other applications. This VBGF has three parameters and is named the "special VBGF" (Pauly 1979). It was later extended to the "generalized VBGF" which has four parameters. Misra (1986) discussed some of the important contributions made by research workers to the knowledge of the VBGF. The special VBGF can easily lead to erroneous results, whereas the large flexibility built into the generalized VBGF enables it to provide a biologically realistic fit to data. The generalized VBGF can be expressed in a number of forms. Misra (1986) considered the following two forms which are extensions of those employed in Misra (1980) for the special VBGF:

$$Y_t^D = \alpha^D + \beta^D * \rho^{Dt} \quad \ldots\ldots\ldots (1)$$

and

$$Y_t^D = \alpha^D (1 - e^{KD(t-t_0)}) \quad \ldots\ldots (2)$$

where

- Y_t is length at age t,
- α is asymptotic maximum length,
- β is the coefficient of regression explained later in the text,
- K is the Brody growth coefficient (Ricker 1975),
- ρ $(=e^{-K})$ is Ford's growth coefficient (Ricker 1975),
- t_0 is the hypothetical age at which $Y_t = 0$ if fish had always grown in the manner described in the equation, and
- D is the surface factor. The value of D varies among groups of fish, usually in the range 0 to 1 although this range may be extended. For the special VBGF, D = 1.

In view of the importance of the generalized VBGF, Misra (1986) developed a statistical method and a computer program (in Fortran) named "GVONB" (a) to fit the generalized VBGF to a sample (a sample comprises several age groups, with one or more observations in an age group), (b) to examine statistically the fit to a sample, and (c) to compare statistically these curves (and their individual parameters) fitted to several samples. Only a brief outline of the computer program was presented in Misra (1986). The objective of this report is to provide a detailed account of the interactive program GVONB and access to it. The report shows how the techniques of sample fitting are implemented and explains how to use the program on an IBM 3081 computer using a VS FORTRAN release 1.4.0 compiler. It summarizes the method of solution, shows overall data flow, diagrams calling sequences between modules, discusses input parameters, provides descriptions of the modules, explains how to run the program, and indicates what outputs are expected. Standard statistical techniques such as ANOVA are indicated but not detailed. The report refers to the listed references to help show where particular calculations are being carried out.

2. SUMMARY OF THE METHODOLOGY

The GVONB program fits the non-linear generalized von Bertalanffy equation (1) to each of several samples of length-at-age data using a series of linear approximations. The program was written bearing in mind that the user may prefer a procedure which first assigns a value to D and then fits the generalized VBGF rather than a procedure where all four parameters are computed simultaneously, for reasons given in Misra (1986). For an assigned value of D, fitting the generalized VBGF to Y-data can be accomplished by any procedure which fits the special VBGF to $Y_1(=Y^D)$-data (Pauly 1979). A method of fitting the special VBGF is discussed in Misra (1980) using techniques detailed in Bliss (1970) and Williams (1967). The program discussed in Misra (1980) was extended in Misra (1986) to the generalized VBGF. Wycove Systems Limited internally documented the program code for the generalized VBGF and modified it to conform more closely to structured programming methods. The code employed by Wycove Systems Limited is ANSI FORTRAN 1977 standards. The following summarizes the discussion of the methods presented in Misra (1980, 1986) and links it to the program input parameters and outputs.

The program GVONB starts with a maximum value DMAX of the range for D, fits equation (1) to the transformed Y_1-data the same way as equation (1) of Misra (1980) was fitted to the Y-data, then decreases D by the value assigned to the variable STEP to a minimum value DMIN, refitting each time while checking for improvements in fit. The user has the option to employ a specific value of D (rather than search for a value from within a range of values) by assigning this value to DMAX and DMIN. It is noted that when D is 1, the program fits the special VBGF. As in the original program, a number of factors can be adjusted by the user to control the precision and range of acceptable values of the parameters of the fit. The techniques used allow fitting of data with age groups unequally spaced and unequal age-group frequencies. Optional methods of weighting the samples can adjust for these non-uniformities caused by heterogeneity of age-group variances. The method also compares the fitted curves and their parameters if the D value for each sample is the same.

The program GVONB begins by calling subroutine INIT that loads user-selected control parameter values into common memory storage. These parameters are discussed in the section "INPUT PARAMETERS AND DATA". The main program then calculates the frequency, mean and mean square for each age group for each sample. This information is then

listed on the results file and is passed to subroutine PLOT that produces lineprinter plots of mean length against age and mean square against age.

Depending on the setting of the parameter NWT, the program then calculates weights for the sample data. If NWT equals 1, subroutine MISRA9 is called to fit a parabola to the logarithm of the sample mean square. The weights used are f_i/antilog(V_i/2), where f_i is the frequency and V_i is the expected value of the log mean square from the parabolic fit. Statistics of the fit are sent to the results file. If NWT equals 2, the weight is the sample frequency divided by the pooled mean square over all samples. If NWT equals 3, the weights are set equal to 1.0 leaving the samples unweighted.

The main program then passes control to MISRA1 which generates the fit to the generalized von Bertalanffy curve for each sample. This is done by a four-level looping structure that is controlled by various of the parameters the user sets in INIT. The outer loop sets and increments the value of D. For each value of D, we can solve the simpler problem as discussed in Misra (1980). For this discussion, rewrite the equation as:

$$Y_{1t} = Y_{1INF} + BETA * RHO^t$$

or

$$Y_{1t} = A + B * R^t.$$

If we replace R^t by Z_t, we see that the equation

$$Y_{1t} = A + B * Z_t$$

is linear in Z. To generate this linear approximation, we need a first estimate of RHO. Then, given this estimate, we need a way to iteratively improve it that will lead to a fit for the nonlinear equation. This problem is discussed in section 16.2 of Bliss (1970). There he shows that if we add variable $Z_{1t} = t * R^{t-1}$ and fit

$$Y_{1t} = A + B * Z_t + B_1 * Z_{1t}$$

then if r was our first estimate of R, $r + b_1/b$ is an improved estimate, where b and b_1 are the computed estimates for B and B_1.

Given that we have an estimate for R, the inside fourth-level loop performs the calculation of b_1/b. If b_1/b is less than a tolerance set in the third-level loop, the fit is checked for acceptance. These checks make sure RHO is less than 1, that the fitted value of Y_{1INF} is less than FCEPT times an initial estimate calculated in the outer loop, and that zero is within TT1(I) standard errors of b_1, where TT1(I) is an input parameter (Student's t) for sample I set for the calculation of confidence limits for B_1, and FCEPT is also an input parameter. If the tolerance is not reached, this loop is run IN2(I) times where IN2(I) is an input parameter.

The third-level loop controls the tolerance used to check b_1/b in the inner loop. It is first set equal to input parameter FACK4(I) and increased by the same amount each iteration. If no fit is found, this loop is run IN3(I) times where IN3(I) is an input parameter.

The second-level loop controls the initial estimate of RHO. There are three routines available for setting RHO: TRY1 uses the Walford line (Ricker 1975), TRY2 uses the first and middle age groups, and TRY3 estimates RHO from the regression of $\log(Y_{1INF} - Y_{1t})$ on t (Bliss 1970, p146). The estimates from TRY2 and TRY3 depend on the estimate of Y_{1INF}. If no fit is found after the third-level loop is finished, Y_{1INF} is increased by input factor FACK(I) and RHO is re-estimated. The number of times this can be done is controlled by input parameter KN1. Since TRY1 does not use Y_{1INF}, the estimate of RHO cannot be altered this way when TRY1 is used, and KN1 is set equal to 1 in that case. The TRY routine to use is selected by input parameter NTRY.

Once we have exited the second-level loop, we have completed the fit for the selected value of D. The best fit is selected on the basis of the R^2 criterion. If a good fit was found, we keep the best one so far and then revise the value of D. If no fit was found, we just revise D. The outer loop controls the setting of the D parameter. It is first set equal to the input parameter DMAX, then reduced each iteration by amount STEP until it is less than DMIN. These are also input parameters. To find the fit for a particular value of D, we can set DMIN = DMAX. Therefore, we can use this program to fit the special VBGF (D=1). After D is set, the program transforms all the data and calls MISRA5 to obtain an initial estimate of Y_{1INF}.

After the outer loop is completed, either a message indicating that no adequate solutions were found is printed or the statistics of the fit are listed. Along with the parameters estimated, the program lists alternate parameters K and t_o for the transformed data (MISRA6) and, for comparison, re-estimates these parameters using Allen's method (MISRA7; see Allen 1966).

If the D values estimated for all samples are the same, MISRA2 is called to estimate a fit for the combined sample. In this routine, the method used is as above but D is not allowed to vary. Control is then returned to the main program.

The main routine finally, if all D's are the same, calls MISRA3 and MISRA4 to compare the samples by ANOVA and chi-square.

3. DATA FLOW

The data flow in and out of the program is quite simple as there are only two sources of inputs and two destinations for outputs (FIG. 1). When the program is run using command procedure GVONB at the terminal, the terminal is connected to unit 5 for input and unit 6 for output. The user is queried through units 5 and 6 for any desired modifications to program control parameters and statistical parameters. The raw data for the generalized von Bertalanffy fit is read from unit 7. The results and messages are written to unit 8. When the command procedure is run, the user is queried for the name of the input file to be attached to unit 7 and the name of the output file to be attached to unit 8. The procedure deletes any file with the same name as the output file before running.

4. INTERNAL STRUCTURE

The Program GVONB contains 28 modules specific to itself and, except for standard Fortran routines, calls no modules from external libraries.

These 28 modules include the main program GVONB, 11 subroutines originally written by Misra that handle the computations, a subroutine PLOT written by Janet Crance for lineprinter plots, a subroutine INIT to obtain user controlled parameters, 4 subroutines from the Scientific Subroutine Package (SSP 1970) to handle matrix manipulations, and 10 functions.

The main logical flow is controlled by the main program. First INIT is called to handle input parameters. Then the program reads the input data, sets up the main data arrays, lists data and plots mean lengths and mean squares. The main program then passes control to MISRA1, which controls the calculation and the statistical test of the fit to the generalized von Bertalanffy curve and, if the samples have compatible parameters, requests a fit for the combined sample. Control is then returned to the main program which, if the curves are compatible, requests an ANOVA and chi-square and re-estimation of error bounds based on pooled mean squares.

Figures 2 to 7 show the calling sequences in GVONB. The module numbers refer to the numbering in the section "MODULE DESCRIPTIONS".

5. COMMON BLOCK DESCRIPTIONS

The program GVONB uses common blocks to pass most parameters and data between subroutines. These common blocks are described in comment lines in the routines. Table 1 shows which routines access each common block.

6. INPUT PARAMETERS AND DATA

There are two types of input to GVONB. The user can interactively change the parameters that control the statistical procedures and set limits on the fitted coefficients. The program reads a file to obtain raw length-at-age data.

6.1 User Controlled Parameters

6.1.1 Parameters Describing Samples

ISTART — Youngest age to use in calculations. Samples of younger ages are ignored.

Default: 2
Range: 1 <= ISTART < N

N — Oldest age to use in calculations. Samples of older ages are ignored.

Default: 28
Range: ISTART < N <= 28

INY — Number of samples.

Default: 2
Range: 1 <= INY <= 9

IYEAR(I) — Code in first two columns of input data, usually last two digits of year sample collected. Used to divide data between samples. INY values needed.

Default: I
Range: 0 < IYEAR(I) <= 99

6.1.2 Parameters used for setting initial values of Y_{1INF} and RHO

FACK1 — Used in MISRA5 as a multiplicative factor for incrementing Y_{1INF}. Y_{1INF} is first set to mean length of oldest age group, then incremented until larger than all mean lengths.

Default: 1.005
Range: FACK1 > 1.0

FACK(I) — A multiplicative factor used to increase Y_{1INF} in TRY2 or TRY3. This is used when re-entering the TRY routine if a satisfactory fit cannot be found for the present estimates of Y_{1INF} and RHO or if RHO derived is greater than 1. INY + 1 values needed, last for combined sample.

Default: 1.05 for 1 <= I <= INY
1.01 for I = INY + 1
Range: FACK(I) > 1

NTRY — This indicates which TRY routine to use.

Default: 3
Range: NTRY = 1, 2, or 3

6.1.3 Parameters used to control maximum number of various iterations in the fitting procedure

IN2(I) — This is the maximum number of times B_1/B, the RHO increment, is calculated before the cutoff value for B_1/B is increased. INY + 1 values.

Default: 10
Range: IN2(I) >= 1

IN3(I) — This is the maximum number of times the B_1/B cutoff is increased before the Y_{1INF} estimate is incremented and RHO is recalculated. INY + 1 values.

Default: 100
Range: IN3(I) >= 1

KN1 — Maximum number of times Y_{1INF} is increased and RHO recalculated before the D value is changed. (For TRY1, KN1 is set to 1 since the method does not depend on the estimate of Y_{1INF}).

Default: 1 if NTRY = 1
0 if NTRY = 2 or 3
Range: KN1 = 1 if NTRY = 1
KN1 >= 1 if NTRY = 2 or 3

6.1.4 Parameters affecting precision of solution

FACK4(I) — Initial value and increment for A1A. Cutoff value used to test RHO increment (B_1/B). If B_1/B < A1A, iterations are ended and the fit is tested. INY + 1 values.

Default: 0.000001
Range: 0 < FACK4(I)

FCEPT — Parameter controlling fitted value of Y_{1INF}. Y_{1INF} must be less than FCEPT times initial estimate of Y_{1INF}.

Default: 1.5
Range: FCEPT >= 1

6.1.5 Values used for selecting D value

DMAX — Maximum (starting) value of D.

Default: 1
Range: 0 < DMAX <= 1

STEP — Step by which D decreases each iteration.

Default: 0.1
Range: 0 < STEP <= DMAX

DMIN — Minimum (ending) value of D. If D is to be held to one value, then DMAX = DMIN.

Default: 0
Range: 0 < DMIN <= DMAX

6.1.6 Other parameters

NWT — Choice of weighting to apply to mean length-at-age data. NWT = 1 for weights based on a parabolic fit of the logarithm of the mean squares. NWT = 2 for weights based on pooled mean square. NWT = 3 for no weighting.

Default: 1
Range: NWT = 1, 2 or 3

TT1(I) — Confidence limits for sample parameters are TT1(I) * standard error. Also used when testing fit. Requires zero to be within TT1(I) standard errors of B_1. INY + 1 values.

Default: TT1(1) = 2.08
TT1(2) = 2.16
TT1(3) = 2.08
Range: TT1(I) > 0

TT — Confidence limits for parameters based on pooled mean squares are TT * standard error.

Default: 2.03
Range: TT > 0

6.2 Input Parameter Summary

Table 2 gives a summary of the input parameters.

6.3 Raw Data Input

The raw data input file is made up of fixed format records giving individual lengths-at-age. These are collected in the program and mean lengths-at-age are calculated. Each input record contains three integer numbers in format I2,I3,I2. The first is the sample code, usually the last two digits of the year, used to

separate the input data by sample. The second is the length and the third is the age of an individual in the sample.

7. MODULE DESCRIPTIONS

1.0 GVONB Main Program

This routine calls INIT to interactively set up input parameters. It then reads in the data, lists the frequencies, mean lengths and mean squares for each age and calls PLOT to produce plots of mean length against age and mean square against age. The routine then generates weights for each sample according to the input parameter NWT (see the section "SUMMARY OF THE METHODOLOGY"). MISRA1 is then called to perform the fits. When control is returned to GVONB, it calls MISRA3 and MISRA4 to compare the samples by ANOVA and chi-square if all D parameters are equal.

2.0 INIT Subroutine

INIT runs an interactive session to accept modifications to input parameters. These parameters are discussed in the section "INPUT PARAMETERS AND DATA".

3.0 PLOT Subroutine

PLOT produces a lineprinter plot of values from a vector passed to it against their subscripts. It is used to plot mean length against age and mean square against age.

4.0 MISRA9 Subroutine

MISRA9 performs a parabolic regression on the log of the sample mean squares. This is used to generate weights for age-group means.

5.0 MISRA1 Subroutine

This subroutine is the core of the program, carrying out the fits of the samples to the generalized von Bertalanffy curve. This routine has a four-level loop structure as follows:

```
initialize variables
    set D estimate to maximum value DMAX (DMAX is
        an input parameter that can be set by the user);

start 1000 loop
    transform data;
    call MISRA5 to provide initial Y_1INF;

    start 990 loop
        call TRY1, TRY2 or TRY3 (depending on input
        parameter NTRY) to provide initial RHO
        (TRY2 or TRY3 automatically increment Y_1INF
        using input factor FACK(I) in case they are
        called again);
        set B_1/B cutoff to 0;

        start 980 loop
            increment B_1/B cutoff by input parameter
            FACK4(I);

            start 970 loop
                calculate B_1/B (see MISRA 1980);
                if B_1/B < tolerance, check fit:
                reject if Y_1INF of fit < 0,
                        if Y_1INF of fit > FCEPT *
                        initial Y_1INF (FCEPT is an
                        input parameter),
                    if estimate of BETA > 0,
                        if 0 not within confidence
                        limits of B_1,
                        if not rejected leave 990
                        loop (goto 999),
                else (if B_1/B >= tolerance)
                        re-estimate RHO = RHO +
                        B_1/B,
                        if RHO > 1, need new
                        estimate of Y_1INF and RHO
                        from TRY routine,
                        goto 990;
970             end of 970 loop, maximum number of
                iterations is controlled by input
                parameter IN2(I);

980         end of 980 loop, maximum number of
            iterations is controlled by input parameter
            IN3(I);

990         end of 990 loop, maximum number of
            iterations is controlled by input parameter
            KN1 (KN1 = 1 if TRY1 is used);
        if finish 990 loop with no fit, then reset D and try
            again (goto 1000);

999 if found a fit, we jump out of all inner loops,
        generate parameters and see if this fit best so
        far (best is decided by looking at R^2); if best,
        save the parameters;
    reset D by subtracting input parameter STEP;
    if new D estimate less than DMIN (input parameter)
        leave loop;

1000 end 1000 loop;
```

MISRA1 then either prints a message indicating no fit was found or generates the statistics for the fit including parameter estimates, confidence limits and an ANOVA of the regression. Calls are made to MISRA6 for alternate parameters K and t_o and to MISRA7 for the same parameters and for Y_{1INF} re-estimated using Allen's method (Allen 1966). If all estimated D values are

equal, MISRA2 is called to calculate a fit for the combined sample. (If DMAX = DMIN only one value of D is used and MISRA2 will be called if any fit meeting the acceptance criteria is found for each sample).

5.1 MISRA5 Subroutine

MISRA5 calculates an initial estimate of Y_{1INF}. It multiplies mean length of oldest group by input parameter FACK1 until larger than all mean lengths.

5.2 TRY1 Subroutine

TRY1 derives an initial value for RHO using the Walford line (see Ricker 1975, p222). This estimate is independent of the initial estimate of Y_{1INF} and hence iteratively increasing Y_{1INF} and re-estimating RHO would not change the fit when using TRY1. Hence, in this case KN1 must be 1.

5.3 TRY2 Subroutine

TRY2 derives an initial value of RHO using the first and middle age groups. The estimate depends on the estimate of Y_{1INF} which is adjusted by input parameter FACK(I) each time the subroutine is called. Hence, iteratively calling TRY2 leads to new starting values for the fitting procedure.

5.4 TRY3 Subroutine

TRY3 derives an initial value of RHO using the regression of $\log(Y_{1INF} - Y_{1t})$ on t. The estimate depends on the estimate of Y_{1INF} which is adjusted by input parameter FACK(I) each time the subroutine is called. Hence, iteratively calling TRY3 leads to new starting values for the fitting procedure.

5.5 MISRA6 Subroutine

The von Bertalanffy growth curve can be written with an alternate set of parameters as in equation (2). This routine derives K and t_o from the fitted parameters for the transformed data.

5.6 MISRA7 Subroutine

MISRA7 re-estimates Y_{1INF}, K and t_o using Allen's method (see Allen 1966).

5.7 MISRA2 Subroutine

MISRA2 fits the generalized von Bertalanffy growth curve to the combined sample. The combined sample is obtained by taking the average mean length at each age. This fit is only calculated if the estimated D value for all samples is equal. This routine uses the same procedure as MISRA1 except that D is fixed at the value obtained for the separate samples.

5.8 Function Subprograms

There are ten function subprograms used to calculate weighted sums. The functions are:

FNX1	weighted sum of Z or Z_1.
FNX2	weighted sum of squares of Z or Z_1 or sum of products of Z and Z_1.
FNY1	weighted sum of Y.
FNY2	weighted sum of squares of Y.
FNXY	weighted sum of products of Y and Z or Z_1.
FNAX	FNX1/sum of weights.
FNAY	FNY1/sum of weights.
FNSSX	FNX2 - (FNX1)(FNX1)/sum of weights for any desired combination of Z and Z_1.
FNSSY	FNY2 - $(FNY1)^2$/sum of weights.
FNSP	FNXY - (FNX1)(FNY1)/sum of weights for Z or Z_1.

5.9 SSP Subroutines

Matrix manipulations are handled by four SSP subroutines (SSP 1970):

SMPY	multiplies a matrix by a scalar.
GMPRD	calculates a matrix product.
LOC	SMPY and GMPRD use the matrices in linear form, LOC calculates the appropriate size of the linear array given the dimensions of the matrix.
MINV	calculates a matrix inverse and determinant.

6.0 MISRA3 Subroutine

MISRA3 compares samples by analysis of variance. It is called only if the estimated D value for all samples is equal.

7.0 MISRA4 Subroutine

MISRA4 generates and outputs confidence limits for sample parameters based on the pooled error mean square. It calculates chi-squares for differences among samples in Y_{1INF}, B and R.

8. HOW TO RUN GVONB

The compiled version of program GVONB is part of the FORT.LOAD partitioned dataset, partition FORT.LOAD(GVONB). There is a command procedure on file CLIST that will run this program. To access this procedure enter:

EX (GVONB)

The computer will respond with a message asking for 'infile'. This is the raw data file as described in the section "INPUT PARAMETERS AND DATA". Enter the file name, for example:

COD3K.DATA

The computer next asks for the 'outfile'. This is the file for results and error messages. The procedure will first delete any file with the same name before the program is run. If there is no file to delete, some informative messages will show up as in the example run in the section "EXAMPLE RUN".

The program will then begin with an interactive session allowing the user to set input parameters. All these parameters are discussed in the section "INPUT PARAMETERS AND DATA". The section "EXAMPLE RUN" contains a sample run. The parameters are divided into six groups:

1. Parameters describing the sample.
2. Parameters for setting Y_{1INF} and RHO.
3. Parameters to control iterations.
4. Parameters affecting precision of solution.
5. Parameters for controlling the D value.
6. Other parameters.

The computer will list a group and their default values on the screen and ask if you want to change one. If you do, enter it's number. The computer will then ask for the new value. When you enter it, the computer will check that it is reasonable. Once all changes are made, enter a zero to go to the next group. The first group of parameters should be checked carefully to be sure they describe the raw data. The other parameters have defaults that should be adequate for most cases. If you want to exit any time, hit the <BREAK> key and you will be returned to the operating system.

Once the input parameters are set, the program will run, with output going to the output file named. When the program terminates, READY is printed on the screen. The output file can then be printed or listed on the terminal.

The source code for the program is stored in file GVONB.SOURCE.DATA and in BACKUP.GVONB.DATA. There is a control file in CNTL that will compile the program and put it in FORT.LOAD(GVONB). This control file and the command procedure GVONB are given in Appendix 1.

9. EXPECTED OUTPUT

Two output units are used by the program. For the interactive session, output comes to the terminal as discussed in the section "HOW TO RUN GVONB" and illustrated in the example run in the next section. The program results and messages go to a unit connected to the 'outfile' named when the GVONB procedure is executed.

Program results will contain the following sections:

1. A table of initial values showing parameters as modified in the interactive session. This gives a record of what parameters are used on the run.

2. A listing of the data for each sample with plots of mean length against age and mean square against age.

3. If weighting is done by fitting a parabola to the log of the mean square, statistics of this fit are listed for each sample.

4. For each sample, the statistics of the best fit are listed including parameters with confidence limits, alternate parameters, an ANOVA of the regression, a listing of observed and predicted values, and the parameters estimated by Allen's method.

5. If all D values are the same, parameters for the combined sample are printed along with a comparison of the samples by ANOVA and chi-square and a re-estimation of the sample parameter confidence limits based on pooled error mean square.

10. EXAMPLE RUN

The example run in Appendix 2 shows a terminal session and a set of outputs. The example was set up to copy the run in Misra (1980). The following input parameters were changed:

ISTART (youngest age) = 1
N (oldest age) = 27

IYEAR(1) (code for 1st sample) = 68
IYEAR(2) (code for 2nd sample) = 40

DMIN (min value for D) = 1

This set of parameters will force D to be 1 and therefore allow comparison of the fits to the two samples.

11. PROGRAM LISTING

The program listing is given in Appendix 3.

The modules are in the following order:

GVONB
INIT
FNX1
FNY1
FNAX
FNAY
FNXY
FNX2
FNY2
FNSSX
FNSSY
FNSP
MISRA1
MISRA2
MISRA3
TRY1
TRY2
TRY3
MISRA4
MISRA5
MISRA6
MISRA7
MISRA9
PLOT
SMPY
GMPRD
LOC
MINV

12. ACKNOWLEDGEMENTS

I am grateful to Miss C. Ervine for reorganising and editing the documentation of the program. The assistance of G. Black and D. Swetnam is acknowledged.

13. REFERENCES

Allen, K. R. 1966. A method of fitting growth curves of the von Bertalanffy type to observed data. J. Fish. Res. Board Can. 23: 163-179.

Bertalanffy, L. von. 1938. A quantitative theory of organic growth. Human Biol. 10: 181-213.

Bertalanffy, L. von. 1960. Principles and theory of growth, p. 137-259. In Wiktor W. Howinski (ed.). Fundamental aspects of normal and malignant growth. Elsevier, Amsterdam.

Bliss, C. E. 1970. Statistics in biology, Vol. 2. McGraw-Hill Book Co., Toronto, Ont. 639 p.

Misra, R. K. 1980. Statistical comparisons of several growth curves of the von Bertalanffy type. Can. J. Fish. Aquat. Sci. 37: 920-926.

Misra, R. K. 1986. Fitting and comparing several growth curves of the generalized von Bertalanffy type. Can. J. Fish. Aquat. Sci. (In Press).

Pauly, D. 1979. Gill size and temperature as governing factors in fish growth: A generalization of von Bertalanffy's growth formula. Ber. Inst. Meereskd. Christian-Albrechts Univ. Kiel(63): 156 p.

Ricker, W. E. 1975. Computation and interpretation of biological statistics of fish populations. Bull. Fish. Res. Board Can. 191: 382 p.

SSP. 1970. System/360 Scientific Subroutine Package, Version III, Programmer's manual. IBM Corporation, New York.

Williams, E. J. 1967. Regression analysis. John Wiley and Sons, Inc., London. 214 p.

TABLE. 1. Common blocks.

Routine	Common Blocks																
	AA	ONER	FNX	FNY	AA1	AA2	AA3	AA4	AA9	AA10	AA11	AA12	AA15	AA19	AA20	COM	CMM
GVONB	X	X	X	X					X	X	X	X	X	X	X		
INIT	X	X	X	X						X	X		X	X	X	X	X
MISRA1	X	X	X	X	X	X	X	X		X	X		X	X		X	X
MISRA2	X	X	X	X	X	X	X	X		X	X		X	X		X	X
MISRA3	X	X	X	X							X					X	
MISRA4		X	X	X												X	X
MISRA5	X	X					X					X					
MISRA6																	
MISRA7	X	X	X	X				X			X				X		X
MISRA9	X	X							X								
TRY1	X	X			X	X				X							
TRY2	X	X			X	X	X			X							
TRY3	X	X			X	X	X			X							
PLOT	X		X	X													
FNX1	X		X														
FNX2	X		X														
FNY1	X			X													
FNY2	X			X													
FNXY	X		X	X													
FNAX	X		X														
FNAY	X			X													
FNSSX	X		X														
FNSSY	X			X													
FNSP	X		X	X													
SMPY																	
MINV																	
GMPRD																	
LOC																	

TABLE. 2. Input parameter summary.

Variable	Number	Default	Range
ISTART	1	2	1 <= ISTART < N
N	1	28	ISTART < N <= 28
INY	1	2	1 <= INY <= 9
IYEAR(I)	INY	I	0 < IYEAR(I) <= 99
FACK1	1	1.005	FACK1 > 1
FACK(I)	INY + 1	1.05 (except for 1.01 for INY + 1)	FACK(I) > 1
NTRY	1	3	NTRY = 1, 2 or 3
IN2(I)	INY + 1	10	IN2(I) >= 1
IN3(I)	INY + 1	100	IN3(I) >= 1
KN1	1	1 if NTRY = 1 10 if NTRY = 2 or 3	KN1 = 1 if NTRY = 1 KN1 >= 1 if NTRY = 2 or 3
FACK4(I)	INY + 1	0.000001	0 < FACK4(I)
FCEPT	1	1.5	FCEPT >= 1
DMAX	1	1	0 < DMAX <= 1
STEP	1	0.1	0 < STEP <= DMAX
DMIN	1	0	0 < DMIN <= DMAX
NWT	1	1	NWT = 1, 2 or 3
TT1(I)	INY + 1	2.08 if I = 1 or 3, 2.16 if I = 2	TT1(I) > 0
TT	1	2.03	TT > 0

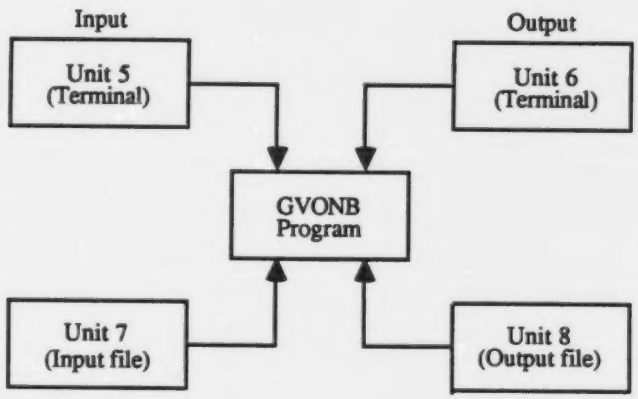


FIG. 1. Data flow diagram for GVONB.

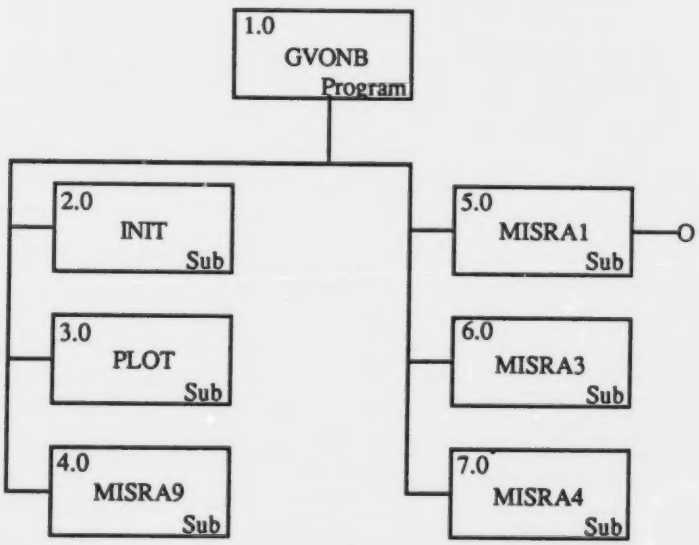


FIG. 2. Internal structure of GVONB program.

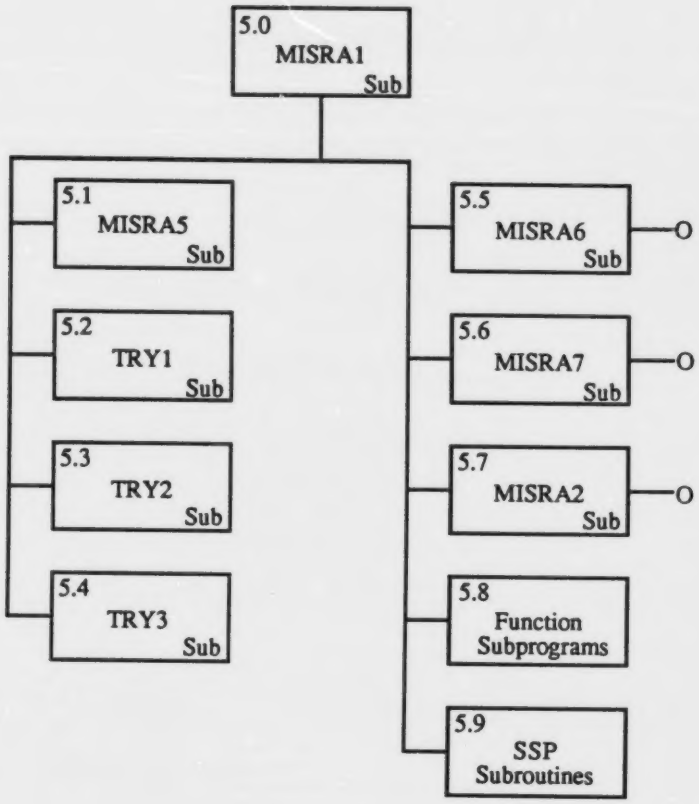


FIG. 3. Internal structure of MISRA1 subroutine.

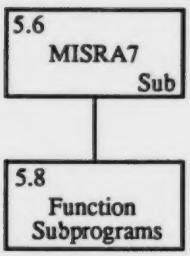


FIG. 4. Internal structure of MISRA7 subroutine.

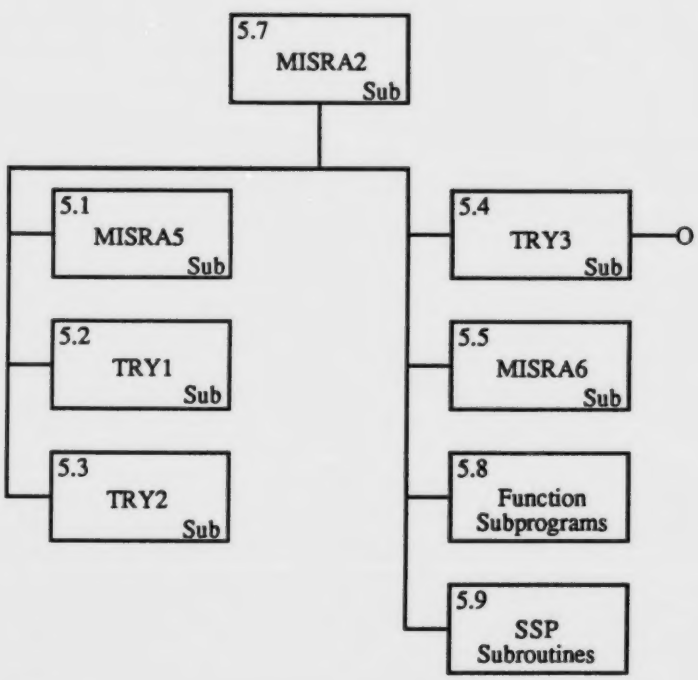


FIG. 5. Internal structure of MISRA2 subroutine.

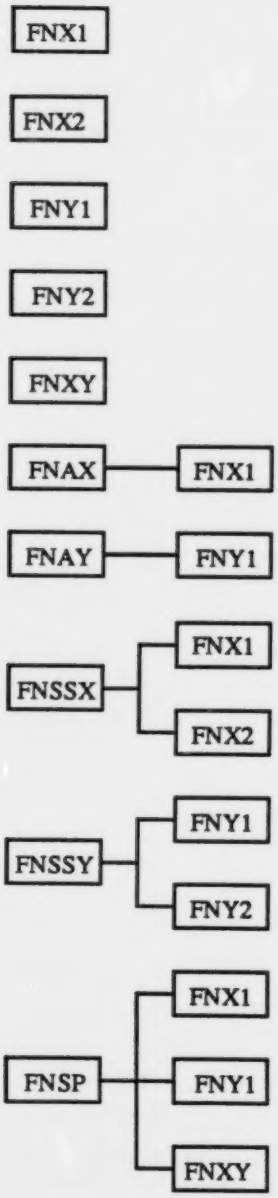


FIG. 6. Internal structure of function subprograms.

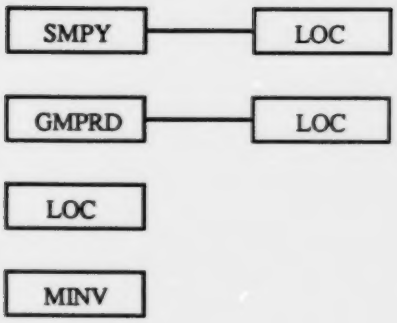


FIG. 7. Internal structure of SSP subroutines.

14. APPENDIX 1

COMPILE AND EXECUTE JOBS

```
L GVONB.COMPL.CNTL
000010 // account information
000020 /*ROUTE  PRINT FETCH
000030 //STEP EXEC FORTVCL.PARM.FORT='SOURCE'
000040 //FORT.SYSIN DD DSN=L24A.GVONB.SOURCE.DATA.DISP=SHR
000050 //LKED.SYSLMOD DD DSN=L24A.FORT.LOAD(GVONB).DISP=SHR
000060 //LKED.SYSIN DD *
AS1510001 END OF DATA
READY

L CLIST(GVONB)
000010  PROC 2 INFILE OUTFILE
000020  ALLOC FI(SYSPRINT) DA(*)
000030  ALLOC FI(SYSTERM)  DA(*)
000040  ALLOC FI(SYSOUT)   DA(*)
000050  FREE  FI(FT05F001)
000060  ALLOC FI(FT05F001) DA(*)
000070  FREE  FI(FT07F001)
000080  ALLOC FI(FT07F001) DA(&INFILE) OLD
000090  FREE  FI(FT06F001)
000100  ALLOC FI(FT06F001) DA(*)
000110  DEL   &OUTFILE
000120  FREE  ATTR(MM)
000130  ATTR  MM BLKSIZE(1330) LRECL(133) RECFM(F B A)
000140  FREE  DA(&OUTFILE)
000150  ALLOC DA(&OUTFILE) USING(MM) SP(5 1) TR RELEASE
000160  FREE  FI(FT08F001)
000170  ALLOC FI(FT08F001) DA(&OUTFILE) OLD
000180  CALL  FORT.LOAD(GVONB)
000190  END
AS1510001 END OF DATA
READY
```

15. APPENDIX 2

EXAMPLE RUN

```
EX (GVONB)
IKJ56700A ENTER POSITIONAL PARAMETER INFILE -
COD3K.DATA
IKJ56700A ENTER POSITIONAL PARAMETER OUTFILE -
OUTPUT2.DATA
IKJ56247I FILE FT05F001 NOT FREED. IS NOT ALLOCATED
IKJ56247I FILE FT07F00I NOT FREED. IS NOT ALLOCATED
IKJ56247I FILE FT06F001 NOT FREED. IS NOT ALLOCATED
IDC3901I ERROR QUALIFYING L24A.OUTPUT2.DATA
IDC3902I ** DEFAULT SERVICE ROUTINE ERROR CODE 20. LOCATE ERROR
CODE 8
IDC0014I LASTCC=8
IKJ56247I FILE MM NOT FREED. IS NOT ALLOCATED
IKJ56247I DATA SET L24A.OUTPUT2.DATA NOT FREED. IS NOT
ALLOCATED
IKJ56247I FILE FT08F001 NOT FREED. IS NOT ALLOCATED


      GVONB PROGRAM

PARAMETERS DESCRIBING SAMPLE
 1 ISTART   =  2
 2 N        = 28
 3 INY      =  2
 4 IYEAR( 1)=   1
 5 IYEAR( 2)=   2

 ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
1
ENTER ISTART
?
1

PARAMETERS DESCRIBING SAMPLE
 1 ISTART   =  1
 2 N        = 28
 3 INY      =  2
 4 IYEAR( 1)=   1
 5 IYEAR( 2)=   2

 ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
2
ENTER N
?
27

PARAMETERS DESCRIBING SAMPLE
 1 ISTART   =  1
 2 N        = 27
 3 INY      =  2
 4 IYEAR( 1)=   1
 5 IYEAR( 2)=   2

 ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
4
```

```
ENTER YEAR( 1)
?
68

PARAMETERS DESCRIBING SAMPLE
 1 ISTART   =  1
 2 N        = 27
 3 INY      =  2
 4 IYEAR( 1)=  68
 5 IYEAR( 2)=   2

 ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
5
ENTER YEAR( 2)
?
40

PARAMETERS DESCRIBING SAMPLE
 1 ISTART   =  1
 2 N        = 27
 3 INY      =  2
 4 IYEAR( 1)=  68
 5 IYEAR( 2)=  40

ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
0

   PARAMETERS FOR SETTING YINF AND RHO
 1 FACK1  =   1.005
 2 FACK(1)=   1.050
 3 FACK(2)=   1.050
 4 FACK(3)=   1.010
 5 NTRY   =  3

 ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
0

    PARAMETERS TO CONTROL ITERATIONS
 1 IN2( 1)=   10
 2 IN2( 2)=   10
 3 IN2( 3)=   10
 4 IN3( 1)=  100
 5 IN3( 2)=  100
 6 IN3( 3)=  100
 7 KN1    =   10

 ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
0

    PARAMETERS AFFECTING PRECISION OF SOLN
 1 FACK4( 1)= 0.00000100
 2 FACK4( 2)= 0.00000100
 3 FACK4( 3)= 0.00000100
 4 FCEPT    =   1.50
```

```
 ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
0

   PARAMETERS FOR CONTROLLING D VALUE
 1 DMAX  =  1.00
 2 STEP  =  0.10
 3 DMIN  =  0.00

ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
0

  OTHER PARAMETERS
 1 NWT    =   1
 2 TT1( 1)=   2.08
 3 TT1( 2)=   2.16
 4 TT1( 3)=   2.08
 5 TT     =   2.03

ENTER NUMBER TO CHANGE. 0 TO CONTINUE
?
0
READY

AS1510000 DSNAME='L24A.OUTPUT.2.DATA'
   TABLE OF INITIAL VALUES


PARAMETERS DESCRIBING SAMPLE

ISTART =  1
     N = 27
    INY = 2
IYEAR ARRAY
 68  40


PARAMETERS USED FOR SETTING INYF AND RHO

FACK1 =  1.005
FACK ARRAY
  1.050   1.050   1.050

NTRY = 3


PARAMETERS TO CONTROL NUMBER OF ITERATIONS

IN2 ARRAY
    10     10     10

IN3 ARRAY
   100    100    100

KN1 =    10
```

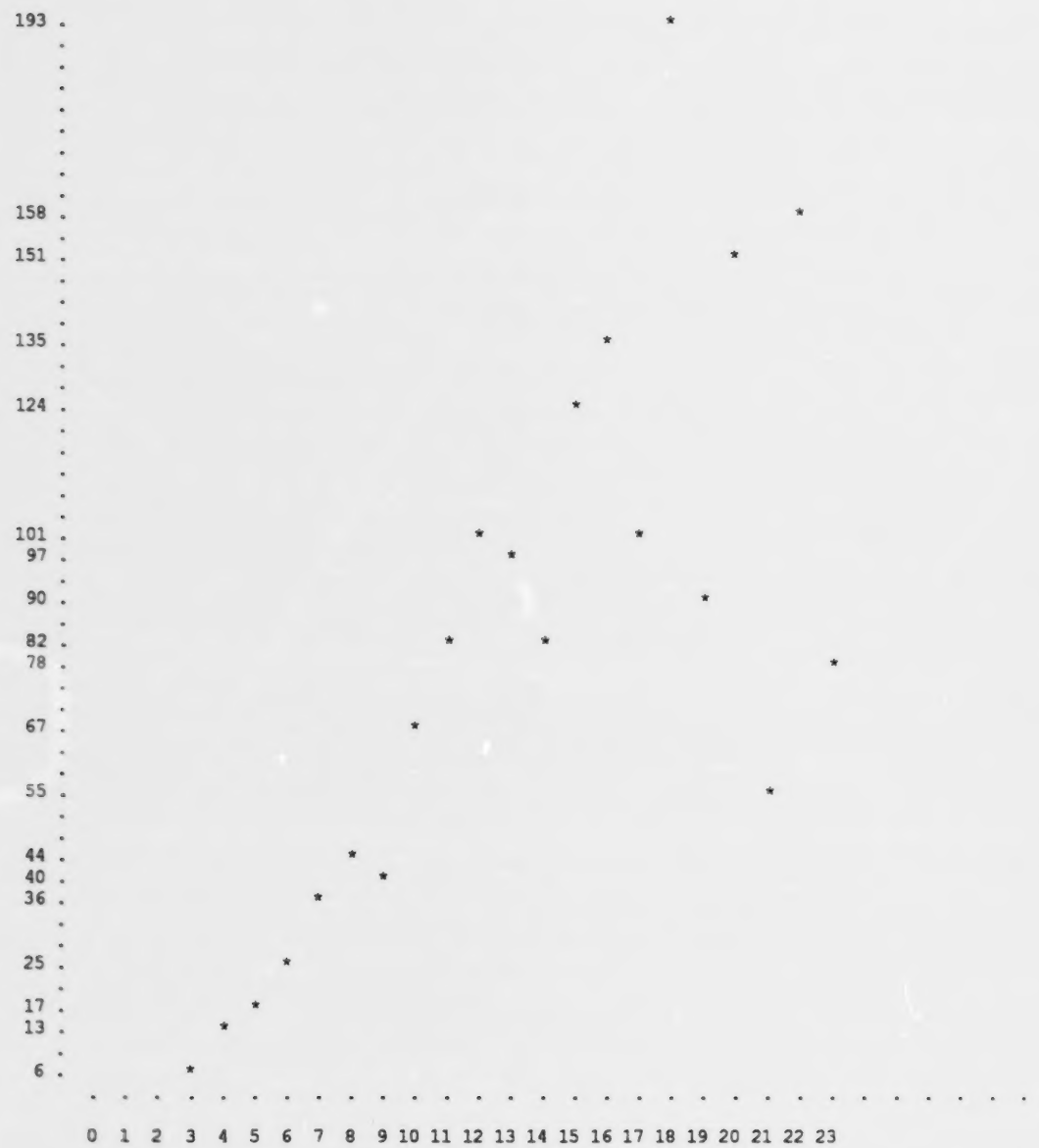
```
PARAMETERS AFFECTING PRECISION OF SOLN

FACK4 ARRAY
.0000010 .0000010 .0000010

FCEPT =  1.50


VALUES USED FOR CONTROLLING D VALUE

DMAX  =  1.00
STEP  =  0.10
DMIN  =  0.00


OTHER PARAMETERS

NWT =  1
TT1 ARRAY
  2.08   2.16   2.08

TT =  2.03


END TABLE OF INITIAL VALUES
```

AS1510000 DSNAME='L24A.OUTPUT2'

SAMPLE	AGE	MEAN LGTH.	DF	MS	GROUP SIZE
1	1				0
1	2				0
1	3	35.3250	79	6.1210	80
1	4	41.2521	356	15.0263	357
1	5	49.0352	595	20.7983	596
1	6	57.4245	489	27.0859	490
1	7	64.4437	301	37.5249	302
1	8	70.4271	191	45.1571	192
1	9	73.1655	144	43.2782	145
1	10	77.1829	81	67.3372	82
1	11	79.9036	82	82.7469	83
1	12	88.0182	54	103.5000	55
1	13	90.2958	70	100.8991	71
1	14	94.5370	53	83.3868	54
1	15	98.5098	50	125.2150	51
1	16	96.7273	32	139.4551	33
1	17	99.9167	11	102.9943	12
1	18	99.7143	13	193.2981	14
1	19	98.3077	12	93.7344	13
1	20	106.1429	6	154.4792	7
1	21	98.0909	10	58.8937	11
1	22	91.3333	2	161.3340	3
1	23	96.0000	2	79.0000	3
1	24				0
1	25	88.0000			1
1	26	98.0000			1
1	27	116.0000			1

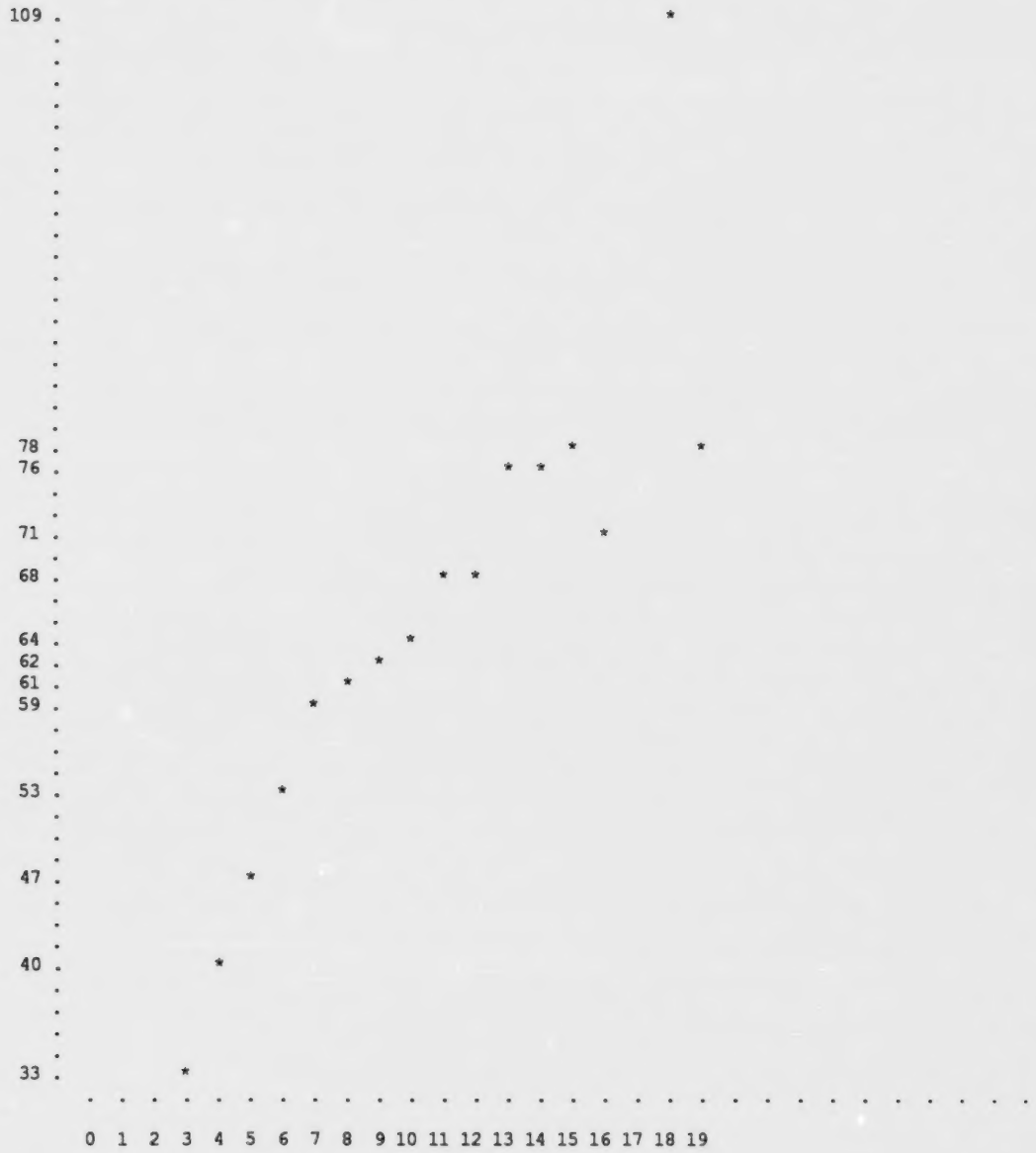
PLOT OF MEAN LENGTH AGAINST AGE

PLOT OF MEAN SQUARE AGAINST AGE



SAMPLE	AGE	MEAN LGTH.	DF	MS	GROUP SIZE
2	1				0
2	2				0
2	3	33.1429	6	34.4766	7
2	4	41.5472	52	10.3690	53
2	5	48.2267	74	22.3674	75
2	6	54.5714	69	25.8723	70
2	7	60.2045	43	30.4462	44
2	8	61.7333	14	15.6381	15
2	9	63.7619	20	18.5906	21
2	10	64.1964	55	22.7068	56
2	11	70.0588	16	66.3125	17
2	12	69.6000	9	84.9336	10
2	13	77.2857	6	88.2383	7
2	14	76.7500	3	37.5833	4
2	15	78.0000			1
2	16	73.0000	1	32.0000	2
2	17				0
2	18	109.0000			1
2	19	78.5000	1	144.5000	2
2	20				0
2	21				0
2	22				0
2	23				0
2	24				0
2	25				0
2	26				0
2	27				0

PLOT OF MEAN LENGTH AGAINST AGE



PLOT OF MEAN SQUARE AGAINST AGE
144
87
84
65
35
32
29
24
21
18
13
10
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

FOR SAMPLE 1

F TEST FOR PARABOLA AND FOR II DEGREE TERMS
FOR PARABOLA: F= 87.971 D.F.= 2 AND 18
FOR II DEGREE TERMS: F= 51.023 D.F.= 1 AND 18

PARABOLA EXPLAINS 90.20 PERCENT OF THE VARIATION

CONSTANTS OF THE PARABOLIC EQUATION ARE:
B(0) = 0.4580 B(1) = 0.4847 B(2) = -0.0133

FOR SAMPLE 2

F TEST FOR PARABOLA AND FOR II DEGREE TERMS
FOR PARABOLA: F= 5.371 D.F.= 2 AND 11
FOR II DEGREE TERMS: F= 0.346 D.F.= 1 AND 11

PARABOLA EXPLAINS 44.81 PERCENT OF THE VARIATION

CONSTANTS OF THE PARABOLIC EQUATION ARE:
B(0) = 2.8356 B(1) = 0.0135 B(2) = 0.0041

FOR SAMPLE 1 (EQUATION 1)

USING TRY3

CONVERGENCE OCCURRED AT 0.0002000 PERCENT INTERVALS

D-FACTOR 0.100000

CONSTANT	VALUE	SE	CONFIDENCE LIMIT LOWER	UPPER
Y-INFINITY	1.59387	0.001031	1.59173	1.596017
SLOPE-B	-0.30359	0.001893	-0.30752	-0.299652
AUXILIARY-B1	0.0000	0.000743	-0.00154	0.001549
RHO	0.82593	0.002446	0.82084	0.831022

FOR THE TRANSFORMED DATA

VALUES DERIVED FROM EQUATION (1) :

Y-INFINITY = 1.593874
K = 0.191240
TIME-ZERO = -8.671008

FOR DATA IN THE ORIGINAL SCALE

VALUES DERIVED FROM EQUATION (1) :

Y-INFINITY = 105.814133
K = 1.912402
TIME-ZERO = -8.671008

ANOVA OF MULTIPLE REGRESSION

ITEM	DF	MS	F VALUE
ALL SLOPES(B AND B1)	2	0.354428	4048.911
SLOPE(B)	1	0.402448	4597.484
AUXILIARY(B1)	1	0.000000	0.000
ERROR	21	0.000088	

BERT.FUNCTION EXPLAINS 99.73 PER CENT
OF THE TOTAL VARIATION AMONG AGES

AT AGE	GROUP SIZE	OBSERVED LENGTH	EXPECTED LENGTH	DIFFERENCE
3	80	1.4283	1.4228	0.0054
4	357	1.4506	1.4526	-0.0020
5	596	1.4759	1.4772	
6	490	1.4994	1.4975	0.0019
7	302	1.5168	1.5143	0.0025
8	192	1.5303	1.5281	0.0022
9	145	1.5361	1.5396	-0.0034
10	82	1.5444	1.5490	-0.0047
11	83	1.5497	1.5568	-0.0071
12	55	1.5648	1.5633	
13	71	1.5688	1.5686	
14	54	1.5760	1.5730	0.0030
15	51	1.5825	1.5766	0.0059
16	33	1.5796	1.5796	
17	12	1.5848	1.5821	0.0026
18	14	1.5844	1.5842	
19	13	1.5822	1.5859	-0.0037
20	7	1.5944	1.5872	0.0071
21	11	1.5818	1.5884	-0.0066
22	3	1.5706	1.5894	-0.0188
23	3	1.5784	1.5901	-0.0117
25	1	1.5648	1.5913	-0.0266
26	1	1.5817	1.5918	-0.0101
27	1	1.6086	1.5921	0.0165

MEAN STANDARDIZED DIFFERENCE IS 5.1759

BY ALLENS ALGORITHM:

CONSTANT		SE
K=	0.194418	0.001018
T-ZERO=	-8.499389	0.057039
Y-INFINITY=	1.592676	0.000367

FOR SAMPLE 2 (EQUATION 1)

USING TRY3

CONVERGENCE OCCURRED AT 0.0009000 PERCENT INTERVALS

D-FACTOR 0.200000

CONSTANT	VALUE	SE	CONFIDENCE LIMIT	
			LOWER	UPPER
Y-INFINITY	2.36695	0.010013	2.34532	2.388582
SLOPE-B	-0.81277	0.051432	-0.92386	-0.701680
AUXILIARY-B1	-0.0000	0.012963	-0.02800	0.027996
RHO	0.75136	0.015949	0.71691	0.785817

FOR THE TRANSFORMED DATA

VALUES DERIVED FROM EQUATION (1) :

Y-INFINITY =	2.366956
K =	0.285861
TIME-ZERO =	-3.739259

FOR DATA IN THE ORIGINAL SCALE

VALUES DERIVED FROM EQUATION (1) :

Y-INFINITY =	74.293777
K =	1.429304
TIME-ZERO =	-3.739259

ANOVA OF MULTIPLE REGRESSION

ITEM	DF	MS	F VALUE
ALL SLOPES(B AND B1)	2	0.249178	214.919
SLOPE(B)	1	0.070500	60.807
AUXILIARY(B1)	1	0.000000	0.000
ERROR	13	0.001159	

BERT.FUNCTION EXPLAINS 96.84 PER CENT
OF THE TOTAL VARIATION AMONG AGES

AT AGE	GROUP SIZE	OBSERVED LENGTH	EXPECTED LENGTH	DIFFERENCE
3	7	2.0141	2.0222	
4	53	2.1072	2.1079	
5	75	2.1710	2.1723	
6	70	2.2253	2.2207	
7	44	2.2695	2.2571	0.0124
8	15	2.2809	2.2844	
9	21	2.2957	2.3049	
10	56	2.2988	2.3203	-0.0215
11	17	2.3393	2.3319	
12	10	2.3363	2.3406	
13	7	2.3857	2.3472	0.0385
14	4	2.3824	2.3521	0.0303
15	1	2.3901	2.3558	0.0343
16	2	2.3587	2.3586	
18	1	2.5556	2.3622	0.1933
19	2	2.3932	2.3634	0.0298

MEAN STANDARDIZED DIFFERENCE IS 3.4451

BY ALLENS ALGORITHM:

	CONSTANT	SE
K=	0.275645	0.047627
T-ZERO=	-3.990088	1.218719
Y-INFINITY=	2.370736	0.020634

AS1510001 END OF DATA

16. APPENDIX 3

PROGRAM LISTING

```
000010 C*************************************************************************
000020 C
000030       PROGRAM GVONB
000040 CC
000050 CC----DESCRIPTION:  PROGRAM TO FIT A GENERALIZED VON BERTALANFFY GROWTH
000060 CC                  CURVE TO EACH OF SEVERAL SAMPLES.  SEE DISCUSSION
000070 CC                  IN R.K. MISRA 1980, 'STATISTICAL COMPARISONS OF
000080 CC                  SEVERAL GROWTH CURVES OF THE VON BERTALANFFY TYPE',
000090 CC                  CAN. J. FISH. AQUAT. SCI. 37:  920-926.  REFERENCE
000100 CC                  IS ALSO MADE TO C.E. BLISS 1970, 'STATISTICS IN
000110 CC                  BIOLOGY', VOL. 2, MCGRAW-HILL, TORONTO; AND K.R.
000120 CC                  ALLEN 1966, 'A METHOD OF FITTING GROWTH CURVES OF
000130 CC                  THE VON BERTALANFFY TYPE TO OBSERVED DATA', J. FISH.
000140 CC                  RES. BOARD CAN. 23:  163-179.
000150 CC
000160 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
000170 CC                       WYCOVE SYSTEMS LIMITED,
000180 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
000190 CC                       RESEARCH BRANCH, DFO, HALIFAX.
000200 CC
000210 CC----DATE:  MARCH 1984.
000220 CC
000230 CC----REVISION HISTORY:  NONE.
000240 CC
000250 CC----FILES:
000260 CC
000270 CC    UNIT    I/O    DESCRIPTION
000280 CC    ------------------------
000290 CC     7       I     INPUT DATA.
000300 CC     8       O     RESULTS AND MESSAGES.
000310 CC
000320 CC----EXTERNALS:
000330 CC
000340 CC    NAME     DESCRIPTION
000350 CC    ------------------
000360 CC    INIT     GETS INPUT PARAMETERS.
000370 CC    PLOT     LINE PRINTER PLOTS.
000380 CC    MISRA1   MAIN CALCULATION SUBROUTINE.
000390 CC    MISRA3   COMPARISON OF SAMPLES BY ANOVA AND CHI-SQUARE.
000400 CC    MISRA4   CONFIDENCE LIMITS BASED ON POOLED SUMS OF SQUARES.
000410 CC    MISRA9   DERIVATION OF WEIGHTS BY PARABOLA FITTED TO SAMPLE MS.
000420 CC
000430 CC----COMMON BLOCKS:
000440 CC
000450 CC    COMMON /AA/
000460 CC
000470 CC    VARIABLE  DESCRIPTION
000480 CC    --------------------
000490 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
000500 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
000510 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
000520 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
000530 CC    N2        N + 1.
000540 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
000550 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
000560 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
000570 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
000580 CC
000590 CC    COMMON /ONER/
000600 CC
000610 CC    VARIABLE  DESCRIPTION
000620 CC    --------------------
000630 CC    IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
000640 CC              ESTIMATE OF 'RHO'.
000650 CC    TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
000660 CC    TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
000670 CC              DATA.
000680 CC    Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
000690 CC              AGE + 1.
000700 CC    DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
000710 CC    N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
000720 CC    SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
```

```
000730 CC     HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
000740 CC     IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
000750 CC               VALUE FOR 'B1/B'.
000760 CC     FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
000770 CC     IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
000780 CC               OF YEAR SAMPLE COLLECTED.
000790 CC     NWT       TYPE OF WEIGHTING:
000800 CC               1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
000810 CC               2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
000820 CC               3 - CONSTANT = 1.
000830 CC     NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
000840 CC               IF 'NTRY'=1 USE 'TRY1', ETC.
000850 CC
000860 CC     COMMON /FNX/
000870 CC
000880 CC     VARIABLE  DESCRIPTION
000890 CC     --------------------
000900 CC     INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
000910 CC     INX1      INX + 1.
000920 CC
000930 CC     COMMON /FNY/
000940 CC
000950 CC     VARIABLE  DESCRIPTION
000960 CC     --------------------
000970 CC     INY       NUMBER OF SAMPLES.
000980 CC     INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
000990 CC
001000 CC     COMMON /AA9/
001010 CC
001020 CC     VARIABLE  DESCRIPTION
001030 CC     --------------------
001040 CC     VMS(10,30)   SAMPLE MEAN SQUARE; IN 'MISRA9' SET TO SQUARE ROOT OF
001050 CC                  PREDICTED SAMPLE MEAN SQUARE.
001060 CC
001070 CC     COMMON /AA10/
001080 CC
001090 CC     VARIABLE  DESCRIPTION
001100 CC     --------------------
001110 CC     FACK(10)  FACTOR FOR INCREASING ESTIMATE OF 'YINF' IN 'TRY2' OR
001120 CC               'TRY3'.
001130 CC     KN1       MAXIMUM NUMBER OF ITERATIONS OF LOOP THAT RE-ESTIMATES
001140 CC               'YINF' BY 'TRY' ROUTINES, THEN REDOES REGRESSION.
001150 CC
001160 CC     COMMON /AA11/
001170 CC
001180 CC     VARIABLE  DESCRIPTION
001190 CC     --------------------
001200 CC     DFAC(10)  'D' VALUE FOR SAMPLE.
001210 CC     DMAX      STARTING (MAXIMUM) VALUE FOR 'D'.
001220 CC     STEP      VALUE SUBTRACTED FROM 'D' EACH ITERATION.
001230 CC     DMIN      MINIMUM VALUE FOR 'D'.
001240 CC     Y4(10,30) TEMPORARY STORAGE FOR 'Y(,)'.
001250 CC
001260 CC     COMMON /AA12/
001270 CC
001280 CC     VARIABLE  DESCRIPTION
001290 CC     --------------------
001300 CC     FACK1     FACTOR USED TO INCREMENT ESTIMATE OF 'YINF' IN 'MISRA5'.
001310 CC
001320 CC     COMMON /AA15/
001330 CC
001340 CC     VARIABLE  DESCRIPTION
001350 CC     --------------------
001360 CC     FCEPT     ACCEPTANCE CONTROL ON 'YINF', MUST BE LESS THAN 'FCEPT'
001370 CC               TIMES REVISED ESTIMATE OF 'YINF' FROM 'TRY' ROUTINES.
001380 CC
001390 CC     COMMON /AA19/
001400 CC
001410 CC     VARIABLE  DESCRIPTION
001420 CC     --------------------
001430 CC     DETC1     MIN VALUE ACCEPTABLE FOR DETERMINANT IF INVERTING
001440 CC               MATRIX.
```

```
001450 CC     DETC2      MAX VALUE ACCEPTABLE FOR DETERMINANT IF INVERTING
001460 CC                MATRIX.
001470 CC     ZFAC       FACTOR FOR SCALING MATRIX IF DETERMINANT TOO BIG OR
001480 CC                SMALL.
001490 CC
001500 CC     COMMON /AA20/
001510 CC
001520 CC     VARIABLE  DESCRIPTION
001530 CC     --------------------
001540 CC     DFACL      BOUNDS USED TO AVOID EXPONENTS CLOSE TO 1.
001550 CC     DFACU      BOUNDS USED TO AVOID EXPONENTS CLOSE TO 1.
001560 CC
001570        COMMON /AA/    ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
001580        COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
001590       >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
001600        COMMON /FNX/   INX,INX1
001610        COMMON /FNY/   INY,INY1
001620        COMMON /AA9/   VMS(10,30)
001630        COMMON /AA10/ FACK(10),KN1
001640        COMMON /AA11/ DFAC(10),DMAX,STEP,DMIN,Y4(10,30)
001650        COMMON /AA12/ FACK1
001660        COMMON /AA15/ FCEPT
001670        COMMON /AA19/ DETC1,DETC2,ZFAC
001680        COMMON /AA20/ DFACL,DFACU
001690 C
001700        DIMENSION Y1(10,30),VSS(10,30),VS1(10,30),VS2(10,30),KHAR(10),
001710       >          N4(30),VMS2(30),Y2(30)
001720 C
001730   600  FORMAT('1')
001740   601  FORMAT(/)
001750   602  FORMAT(//)
001760 C
001770 C
001780 C
001790 C-----INITIALIZE STORAGE VARIABLES.
001800        DO 781 I=1,INY1
001810            LNY(I)=0
001820            HARN1(I)=0.
001830            KHAR(I)=0
001840            DO 781 J=1,N2
001850                N1(I,J)=0
001860                Y(I,J)=0.
001870                Y4(I,J)=0.
001880                VS1(I,J)=0.
001890                VS2(I,J)=0.
001900   781  CONTINUE
001910        SUMVSS=0.
001920        IVSSDF=0
001930 C
001940 C-----INTERACTIVE INITIALIZATION ROUTINE.
001950        CALL INIT
001960 C
001970 C-----DATA INPUT LOOP.  STORE BY SAMPLE AND AGE.  ADD 1 TO AGE TO HANDLE
001980 C     ZERO AGE FISH.  SKIP FISH BELOW YOUNGEST AND OVER OLDEST AGES
001990 C     WANTED.  SKIP IF WRONG SAMPLE CODE.
002000        DO 863 INP=1,100000
002010            READ(7,505,END=864)JYEAR,JLN,JAGE
002020   505      FORMAT(I2,I3,I2)
002030            JAGE=JAGE+1
002040            IF(JAGE.LT.ISTART.OR.JAGE.GT.N) GOTO 863
002050            AJLN=JLN
002060            DO 786 J=1,INY
002070                IF(JYEAR.EQ.IYEAR(J)) THEN
002080                    JYEAR=J
002090                    GOTO 787
002100                ENDIF
002110   786      CONTINUE
002120            GOTO 863
002130   787      N1(JYEAR,JAGE)=N1(JYEAR,JAGE)+1
002140            VS1(JYEAR,JAGE)=VS1(JYEAR,JAGE)+AJLN
002150            VS2(JYEAR,JAGE)=VS2(JYEAR,JAGE)+AJLN**2
002160   863  CONTINUE
```

```
002170  864  DO 797 K=1,INY1
002180         DO 797 I=ISTART,N2
002190           VMS(K,I)=0.
002200           W3(K,I)=0.
002210           W1(K,I)=0.
002220  797  CONTINUE
002230 C
002240 C
002250 C-----CALCULATE MEANS AND VARIANCES.  LIST INPUTS.  PLOT MEANS AND MS
002260 C     BY AGE FOR EACH SAMPLE.
002270       DO 748 K=1,INY
002280           WRITE(8,645)
002290  645      FORMAT(///,1X,'SAMPLE',T12,'AGE',T21,'MEAN LGTH.',
002300      >           T33,'DF',T43,'MS',T51,'GROUP SIZE')
002310           WRITE(8,601)
002320           DO 706 I=ISTART,N
002330               I2=I-1
002340               IF(N1(K,I).EQ.0) THEN
002350                   WRITE(8,3039) K,I2,N1(K,I)
002360  3039             FORMAT(3X,I2,T11,I3,T51,I4)
002370                 ELSE
002380                   J1=N1(K,I)
002390                   J2=J1-1
002400                   Y(K,I)=VS1(K,I)/FLOAT(J1)
002410                   Y4(K,I)=Y(K,I)
002420                   IF(J1.EQ.1) THEN
002430                       W1(K,I)=1.
002440                       WRITE(8,3036) K,I2,Y(K,I),N1(K,I)
002450  3036                 FORMAT(3X,I2,T11,I3,T18,F10.4,T51,I4)
002460                     ELSE
002470                       VSS(K,I)=VS2(K,I)-VS1(K,I)**2/FLOAT(J1)
002480                       SUMVSS=SUMVSS+VSS(K,I)
002490                       IVSSDF=IVSSDF+J1-1
002500                       VMS(K,I)=VSS(K,I)/FLOAT(J2)
002510                       W1(K,I)=FLOAT(J1)/VMS(K,I)
002520                       WRITE(8,646)K,I2,Y(K,I),J2,VMS(K,I),J1
002530  646                  FORMAT(3X,I2,T11,I3,T18,F10.4,T31,I4,T37,F10.4
002540      >                       ,T51,I4)
002550                   ENDIF
002560               ENDIF
002570  706      CONTINUE
002580           WRITE(8,990)
002590  990      FORMAT(///,17X,'PLOT OF MEAN LENGTH AGAINST AGE',//)
002600           CALL PLOT(K,Y)
002610           WRITE(8,991)
002620  991      FORMAT(///,17X,'PLOT OF MEAN SQUARE AGAINST AGE',//)
002630           CALL PLOT(K,VMS)
002640  748  CONTINUE
002650 C
002660 C
002670 C-----NOW GET WEIGHTS ACCORDING TO VALUE OF 'NWT':
002680 C                1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
002690 C                2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
002700 C                3 - CONSTANT = 1.
002710       IF (NWT.EQ.1) THEN
002720           WRITE(8,600)
002730           DO 794 K=1,INY
002740               DO 3037 I=1,N
002750                   Y2(I)=0.
002760  3037         CONTINUE
002770               DO 793 I=ISTART,N
002780                   VMS2(I)=VMS(K,I)
002790                   Y2(I)=I
002800  793          CONTINUE
002810               CALL MISRA9(K,Y2,VMS2,N)
002820  794      CONTINUE
002830       ELSE IF (NWT.EQ.2) THEN
002840           DO 791 K=1,INY
002850               DO 792 I=ISTART,N
002860                   IF (N1(K,I).NE.0) THEN
002870                       J1=N1(K,I)
002880                       VMS(K,I)=SUMVSS/IVSSDF
```

```
002890                        W1(K,I)=J1/VMS(K,I)
002900                      ENDIF
002910  792               CONTINUE
002920  791           CONTINUE
002930            ELSE
002940               DO 879 K=1,INY1
002950                  DO 878 I=ISTART,N2
002960                     IF (N1(K,I).NE.0) W1(K,I)=1.0
002970  878             CONTINUE
002980  879          CONTINUE
002990            ENDIF
003000 C
003010 C
003020 C-----CALCULATE HARMONIC MEAN FREQUENCY 'HARN1()', WEIGHTS FOR AGE,
003030 C     AND SAMPLE TOTALS.
003040            DO 785 J=ISTART,N
003050              DO 785 I=1,INY
003060                N1(INY1,J)=N1(INY1,J)+N1(I,J)
003070  785       CONTINUE
003080            DO 764 K=1,INY
003090               DO 765 I=ISTART,N
003100                  IF(N1(K,I).EQ.0) THEN
003110                     KHAR(K)=KHAR(K)+1
003120                   ELSE
003130                     HARN1(K)=HARN1(K)+1./FLOAT(N1(K,I))
003140                  ENDIF
003150  765          CONTINUE
003160               KHAR(K)=N-ISTART+1-KHAR(K)
003170               HARN1(K)=KHAR(K)/HARN1(K)
003180  764       CONTINUE
003190            DO 725 I=ISTART,N
003200               LNA(I)=0
003210               IN=INY
003220               DO 726 J=1,INY
003230                  IF(N1(J,I).EQ.0) THEN
003240                     LNA(I)=LNA(I)+1
003250                     IN=IN-1
003260                   ELSE
003270                     W1(INY1,I)=W1(INY1,I)+1./W1(J,I)
003280                  ENDIF
003290  726          CONTINUE
003300               IF(IN.EQ.0) THEN
003310                  LNY(INY1)=LNY(INY1)+1
003320                ELSE
003330                  W1(INY1,I)=IN**2/W1(INY1,I)
003340               ENDIF
003350  725       CONTINUE
003360            DO 772 I=1,N2
003370               N4(I)=0
003380  772       CONTINUE
003390            DO 766 I=ISTART,N
003400               DO 768 J=1,INY
003410                  N4(I)=N4(I)+N1(J,I)
003420  768          CONTINUE
003430               IF(N4(I).EQ.0) THEN
003440                  KHAR(INY1)=KHAR(INY1)+1
003450                ELSE
003460                  HARN1(INY1)=HARN1(INY1)+1./FLOAT(N4(I))
003470               ENDIF
003480  766       CONTINUE
003490            KHAR(INY1)=N-ISTART+1-KHAR(INY1)
003500            HARN1(INY1)=KHAR(INY1)/HARN1(INY1)
003510            DO 736 I=1,INY
003520               IN=N-ISTART+1
003530               DO 737 J=ISTART,N
003540                  IF(N1(I,J).EQ.0) THEN
003550                     LNY(I)=LNY(I)+1
003560                     IN=IN-1
003570                   ELSE
003580                     W1(I,N2)=W1(I,N2)+1./W1(I,J)
003590                  ENDIF
003600  737          CONTINUE
```

```
003610           IF(IN.NE.0) W1(I,N2)=IN**2/W1(I,N2)
003620  736   CONTINUE
003630        DO 789 J=1,INY1
003640          DO 789 J1=ISTART,N2
003650            W3(J,J1)=W1(J,J1)
003660  789   CONTINUE
003670 C
003680 C
003690 C-----TRANSFER TO 'MISRA1' TO PERFORM THE REGRESSION, THEN, IF MORE
003700 C     THAN ONE SAMPLE AND ALL HAVE THE SAME 'D' VALUES, COMPARE THEM
003710 C     WITH ANOVA AND CHI-SQUARE ('MISRA3' AND 'MISRA4').
003720        CALL MISRA1
003730        IF(INY.GT.1) THEN
003740            A1=DFAC(1)
003750            DO 303 I=2,INY
003760                IF(DFAC(I).NE.A1)GOTO 3080
003770  303       CONTINUE
003780            CALL MISRA3
003790            CALL MISRA4
003800        ENDIF
003810  3080  STOP
003820        END
003830        SUBROUTINE INIT
003840 CC
003850 CC----DESCRIPTION:  SUBROUTINE TO PROMPT THE USER FOR INPUT PARAMETERS.
003860 CC
003870 CC----ORIGIN:  WRITTEN BY MARCELLA MILLIGAN,
003880 CC                        WYCOVE SYSTEMS LIMITED,
003890 CC                    FOR DR. R.K. MISRA,
003900 CC                        RESEARCH BRANCH, DFO, HALIFAX.
003910 CC
003920 CC----DATE:  MARCH 1984.
003930 CC
003940 CC----REVISION HISTORY:  NONE.
003950 CC
003960 CC----FILES:
003970 CC
003980 CC     UNIT  I/O  DESCRIPTION
003990 CC     --------------------
004000 CC      5     I   TERMINAL INPUT.
004010 CC      6     O   TERMINAL OUTPUT.
004020 CC
004030 CC----COMMON BLOCKS:
004040 CC
004050 CC     COMMON /AA/
004060 CC
004070 CC     VARIABLE  DESCRIPTION
004080 CC     -------------------
004090 CC     ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
004100 CC               THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
004110 CC     N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
004120 CC               THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
004130 CC     N2        N + 1.
004140 CC     W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
004150 CC     W3(10,30) TEMPORARY COPY OF 'W1(,)'.
004160 CC     LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
004170 CC     LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
004180 CC
004190 CC     COMMON /ONER/
004200 CC
004210 CC     VARIABLE  DESCRIPTION
004220 CC     -------------------
004230 CC     IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
004240 CC               ESTIMATE OF 'RHO'.
004250 CC     TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
004260 CC     TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
004270 CC               DATA.
004280 CC     Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
004290 CC               AGE + 1.
004300 CC     DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
004310 CC     N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
004320 CC     SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
```

```
004330 CC     HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
004340 CC     IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
004350 CC               VALUE FOR 'B1/B'.
004360 CC     FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
004370 CC     IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
004380 CC               OF YEAR SAMPLE COLLECTED.
004390 CC     NWT       TYPE OF WEIGHTING:
004400 CC               1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
004410 CC               2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
004420 CC               3 - CONSTANT = 1.
004430 CC     NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
004440 CC               IF 'NTRY'=1 USE 'TRY1', ETC.
004450 CC
004460 CC     COMMON /FNX/
004470 CC
004480 CC     VARIABLE  DESCRIPTION
004490 CC     --------------------
004500 CC     INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
004510 CC     INX1      INX + 1.
004520 CC
004530 CC     COMMON /FNY/
004540 CC
004550 CC     VARIABLE  DESCRIPTION
004560 CC     --------------------
004570 CC     INY       NUMBER OF SAMPLES.
004580 CC     INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
004590 CC
004600 CC     COMMON /AA10/
004610 CC
004620 CC     VARIABLE  DESCRIPTION
004630 CC     --------------------
004640 CC     FACK(10)  FACTOR FOR INCREASING ESTIMATE OF 'YINF' IN 'TRY2' OR
004650 CC               'TRY3'.
004660 CC     KN1       MAXIMUM NUMBER OF ITERATIONS OF LOOP THAT RE-ESTIMATES
004670 CC               'YINF' BY 'TRY' ROUTINES, THEN REDOES REGRESSION.
004680 CC
004690 CC     COMMON /AA11/
004700 CC
004710 CC     VARIABLE  DESCRIPTION
004720 CC     --------------------
004730 CC     DFAC(10)  'D' VALUE FOR SAMPLE.
004740 CC     DMAX      STARTING (MAXIMUM) VALUE FOR 'D'.
004750 CC     STEP      VALUE SUBTRACTED FROM 'D' EACH ITERATION.
004760 CC     DMIN      MINIMUM VALUE FOR 'D'.
004770 CC     Y4(10,30) TEMPORARY STORAGE FOR 'Y(,)'.
004780 CC
004790 CC     COMMON /AA12/
004800 CC
004810 CC     VARIABLE  DESCRIPTION
004820 CC     --------------------
004830 CC     FACK1     FACTOR USED TO INCREMENT ESTIMATE OF 'YINF' IN 'MISRA5'.
004840 CC
004850 CC     COMMON /AA15/
004860 CC
004870 CC     VARIABLE  DESCRIPTION
004880 CC     --------------------
004890 CC     FCEPT     ACCEPTANCE CONTROL ON 'YINF', MUST BE LESS THAN 'FCEPT'
004900 CC               TIMES REVISED ESTIMATE OF 'YINF' FROM 'TRY' ROUTINES.
004910 CC
004920 CC     COMMON /AA19/
004930 CC
004940 CC     VARIABLE  DESCRIPTION
004950 CC     --------------------
004960 CC     DETC1     MIN VALUE ACCEPTABLE FOR DETERMINANT IF INVERTING
004970 CC               MATRIX.
004980 CC     DETC2     MAX VALUE ACCEPTABLE FOR DETERMINANT IF INVERTING
004990 CC               MATRIX.
005000 CC     ZFAC      FACTOR FOR SCALING MATRIX IF DETERMINANT TOO BIG OR
005010 CC               SMALL.
005020 CC
005030 CC     COMMON /AA20/
005040 CC
```

```
005050 CC     VARIABLE  DESCRIPTION
005060 CC     --------------------
005070 CC     DFACL     BOUNDS USED TO AVOID EXPONENTS CLOSE TO 1.
005080 CC     DFACU     BOUNDS USED TO AVOID EXPONENTS CLOSE TO 1.
005090 CC
005100 CC----LOCAL VARIABLES AND PARAMETERS:
005110 CC
005120 CC     VARIABLE  DESCRIPTION
005130 CC     --------------------
005140 CC     NUM       NUMBER OF PARAMETER TO CHANGE.
005150 CC
005160 CC
005170 C
005180        COMMON /AA/   ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
005190        COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
005200       >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
005210        COMMON /FNY/  INY,INY1
005220        COMMON /AA10/ FACK(10),KN1
005230        COMMON /AA11/ DFAC(10),DMAX,STEP,DMIN,Y4(10,30)
005240        COMMON /AA12/ FACK1
005250        COMMON /AA15/ FCEPT
005260        COMMON /FNX/  INX,INX1
005270        COMMON /AA19/ DETC1,DETC2,ZFAC
005280        COMMON /AA20/ DFACL,DFACU
005290 C
005300 C-----PRINT HEADER.
005310        WRITE(6,'(1H ,///5X,A)') '  GVONB PROGRAM '
005320 C
005330 C-----INITIALIZE VARIABLES.
005340 C
005350        ISTART=2
005360        N=28
005370        INY=2
005380        INY1=INY+1
005390        DO 10 I=1,INY
005400            IYEAR(I)=I
005410  10    CONTINUE
005420 C
005430 C-----WRITE PARAMETERS.
005440  15    CONTINUE
005450        WRITE(6,'(/,1H ,A)') 'PARAMETERS DESCRIBING SAMPLE'
005460 C
005470        WRITE(6,'(1H ,A,I2)') ' 1 ISTART   = ',ISTART
005480        WRITE(6,'(1H ,A,I2)') ' 2 N        = ',N
005490        WRITE(6,'(1H ,A,I2)') ' 3 INY      = ',INY
005500        DO 18 I=1,INY
005510            WRITE(6,'(1H ,I2,A,I2,A,I3)') I+3,' IYEAR(',I,')= ',IYEAR(I)
005520  18    CONTINUE
005530 C
005540 C-----ASK FOR INPUT.
005550  19    WRITE(6,'(/,A)') '  ENTER NUMBER TO CHANGE, 0 TO CONTINUE'
005560        READ (5,*,END=19,ERR=19) NUM
005570 C
005580 C-----PROCESS ACCORDING TO NUMBER.
005590        IF (NUM.EQ.1) THEN
005600  20        WRITE(6,'(1H ,A)') 'ENTER ISTART'
005610            READ(5,*,ERR=20,END=20) ISTART
005620            IF (ISTART.LT.1) THEN
005630                WRITE(6,'(1H ,A)') '  ENTER >= 1'
005640                GOTO 20
005650            ENDIF
005660          ELSE IF (NUM.EQ.2) THEN
005670  22        WRITE(6,'(1H ,A)') 'ENTER N'
005680            READ(5,*,ERR=22,END=22) N
005690            IF (N.LE.ISTART.OR.N.GE.29) THEN
005700                WRITE(6,'(1H ,A)') '  ENTER ISTART < N <= 28'
005710                GOTO 22
005720            ENDIF
005730          ELSE IF (NUM.EQ.3) THEN
005740  24        WRITE(6,'(1H ,A)') 'ENTER INY'
005750            READ(5,*,END=24,ERR=24) INY
005760            IF (INY.LT.1.OR.INY.GT.9) THEN
```

```
005770                WRITE(6,'(1H ,A)') '  ENTER 1 TO 9'
005780                GOTO 24
005790              ENDIF
005800              INY1=INY+1
005810            ELSE IF (NUM.GT.3.AND.NUM.LE.INY+3) THEN
005820  26          WRITE(6,'(1H ,A,I2,A)') 'ENTER YEAR(',NUM-3,')'
005830              READ(5,*,ERR=26,END=26) IYEAR(NUM-3)
005840              IF (IYEAR(NUM-3).LT.0.OR.IYEAR(NUM-3).GT.99) THEN
005850                  WRITE(6,'(1H ,A)') '  ENTER 0 TO 99'
005860                  GOTO 26
005870              ENDIF
005880          ENDIF
005890          IF (NUM.NE.0) GOTO 15
005900          ISTART=ISTART+1
005910          N=N+1
005920 C
005930 C-----INITIALIZE PARAMETERS.
005940          FACK1=1.005
005950          DO 30 I=1,INY
005960              FACK(I)=1.05
005970   30     CONTINUE
005980          FACK(INY1)=1.01
005990          NTRY=3
006000 C
006010 C-----PRINT PARAMETERS.
006020   32     CONTINUE
006030          WRITE(6,'(/A)') '    PARAMETERS FOR SETTING YINF AND RHO'
006040 C
006050          WRITE(6,'(1H ,A,F7.3)') ' 1 FACK1  = ',FACK1
006060          DO 34 I=1,INY1
006070              WRITE(6,'(I3,A,I1,A,F7.3)') I+1,' FACK(',I,')= ',FACK(I)
006080   34     CONTINUE
006090          WRITE(6,'(1H ,I2,A,I1)') INY1+2,' NTRY   = ',NTRY
006100 C
006110 C-----ASK FOR INPUT.
006120   35     WRITE(6,'(/,A)') '  ENTER NUMBER TO CHANGE, 0 TO CONTINUE'
006130          READ (5,*,ERR=35,END=35) NUM
006140 C
006150          IF (NUM.EQ.1) THEN
006160   36         WRITE(6,'(1H ,A)') 'ENTER FACK1'
006170              READ(5,*,ERR=36,END=36) FACK1
006180              IF (FACK1.LE.1.0) THEN
006190                  WRITE(6,'(1H ,A)') '  ENTER > 1.0'
006200                  GOTO 36
006210              ENDIF
006220            ELSE IF (NUM.GT.1.AND.NUM.LE.1+INY1) THEN
006230   38         WRITE(6,'(1H ,A,I2,A)') 'ENTER FACK(',NUM-1,')'
006240              READ(5,*,ERR=38,END=38) FACK(NUM-1)
006250              IF (FACK(NUM-1).LE.1) THEN
006260                  WRITE(6,'(1H ,A)') '  ENTER > 1.0'
006270                  GOTO 38
006280              ENDIF
006290            ELSE IF (NUM.EQ.2+INY1) THEN
006300   40         WRITE(6,'(1H ,A)') '  ENTER NTRY'
006310              READ(5,*,ERR=40,END=40) NTRY
006320              IF (NTRY.LT.1.OR.NTRY.GT.3) THEN
006330                  WRITE(6,'(1H ,A)') '  ENTER 1, 2, OR 3'
006340                  GOTO 40
006350              ENDIF
006360          ENDIF
006370          IF (NUM.NE.0) GOTO 32
006380 C
006390 C-----INITIALIZE PARAMETERS.
006400          DO 50 I=1,INY1
006410              IN2(I)=10
006420              IN3(I)=100
006430   50     CONTINUE
006440          IF (NTRY.EQ.1) THEN
006450              KN1=1
006460            ELSE
006470              KN1=10
006480          ENDIF
```

```
006490 C
006500 C-----WRITE PARAMETERS.
006510  52   CONTINUE
006520       WRITE(6,'(/1H ,A)') '    PARAMETERS TO CONTROL ITERATIONS'
006530       DO 54 I=1,INY1
006540           WRITE(6,'(1H ,I2,A,I2,A,I4)') I,' IN2(',I,')= ',IN2(I)
006550  54   CONTINUE
006560       DO 56 I=1,INY1
006570           WRITE(6,'(1H ,I2,A,I2,A,I4)') I+INY1,' IN3(',I,')= ',IN3(I)
006580  56   CONTINUE
006590       WRITE(6,'(I3,A,I4)') 1+2*INY1,' KN1    = ',KN1
006600 C
006610 C-----ASK FOR CHANGES.
006620  58   WRITE(6,'(/1H ,A)') ' ENTER NUMBER TO CHANGE, 0 TO CONTINUE'
006630 C
006640       READ(5,*,END=58,ERR=58) NUM
006650       IF (NUM.GT.0.AND.NUM.LE.INY1) THEN
006660  60       WRITE(6,'(1H ,A,I2,A)') ' ENTER IN2(',NUM,')'
006670           READ(5,*,ERR=60,END=60) IN2(NUM)
006680           IF (IN2(NUM).LT.1.0) THEN
006690               WRITE(6,'(1H ,A)') '  ENTER >= 1.0'
006700               GOTO 60
006710           ENDIF
006720         ELSE IF (NUM.GT.INY1.AND.NUM.LE.2*INY1) THEN
006730  62       WRITE(6,'(1H ,A,I2,A)') ' ENTER IN3(',NUM-INY1,')'
006740           READ(5,*,ERR=62,END=62) IN3(NUM-INY1)
006750           IF (IN3(NUM-INY1).LT.1.0) THEN
006760               WRITE(6,'(1H ,A)') '  ENTER >= 1.0'
006770               GOTO 62
006780           ENDIF
006790         ELSE IF (NUM.EQ.1+2*INY1) THEN
006800           IF (NTRY.EQ.1) THEN
006810               WRITE(6,'(1H ,A)') ' CANNOT CHANGE KN1 WHEN NTRY=1'
006820             ELSE
006830  64           WRITE(6,'(1H ,A)') ' ENTER KN1'
006840               READ(5,*,ERR=64,END=64) KN1
006850               IF (KN1.LT.1.0) THEN
006860                   WRITE(6,'(1H ,A)') ' ENTER >= 1.0'
006870                   GOTO 64
006880               ENDIF
006890           ENDIF
006900       ENDIF
006910       IF (NUM.NE.0) GOTO 52
006920 C
006930 C
006940 C-----INITIALIZE PARAMETERS.
006950       DO 70 I=1,INY1
006960           FACK4(I)=.000001
006970  70   CONTINUE
006980       FCEPT=1.5
006990 C
007000 C-----WRITE PARAMETERS.
007010  72   CONTINUE
007020       WRITE(6,'(/1H ,A)') '    PARAMETERS AFFECTING PRECISION OF SOLN'
007030 C
007040       DO 74 I=1,INY1
007050           WRITE(6,'(1H ,I2,A,I2,A,F10.8)') I,' FACK4(',I,')= ',FACK4(I)
007060  74   CONTINUE
007070       WRITE(6,'(1H ,I2,A,F6.2)') INY1+1,' FCEPT    = ',FCEPT
007080 C
007090 C-----ASK FOR CHANGES.
007100  76   WRITE(6,'(/1H ,A)') ' ENTER NUMBER TO CHANGE, 0 TO CONTINUE'
007110 C
007120       READ(5,*,ERR=76,END=76) NUM
007130       IF (NUM.GT.0.AND.NUM.LE.INY1) THEN
007140  78       WRITE(6,'(1H ,A,I2,A)') '  ENTER FACK4(',NUM,')'
007150           READ(5,*,ERR=78,END=78) FACK4(NUM)
007160           IF (FACK4(NUM).LE.0.0.OR.FACK4(NUM).GE.1.0) THEN
007170               WRITE(6,'(1H ,A)') '  ENTER 0 < FACK4 < 1'
007180               GOTO 78
007190           ENDIF
007200         ELSE IF (NUM.EQ.INY1+1) THEN
```

```
007210   80      WRITE(6,'(1H ,A)') '  ENTER FCEPT'
007220           READ(5,*,ERR=80,END=80) FCEPT
007230           IF (FCEPT.LT.1.0) THEN
007240               WRITE(6,'(1H ,A)') '  ENTER >= 1.0'
007250               GOTO 80
007260           ENDIF
007270       ENDIF
007280       IF (NUM.NE.0) GOTO 72
007290 C
007300 C-----INITIALIZE PARAMETERS FOR CONTROLLING 'D' VALUE.
007310       DMAX=1.0
007320       STEP=0.1
007330       DMIN=0.0
007340 C
007350 C-----WRITE PARAMETERS.
007360   90  CONTINUE
007370       WRITE(6,'(/1H ,A)') '   PARAMETERS FOR CONTROLLING D VALUE'
007380 C
007390       WRITE(6,'(1H ,A,F5.2)') ' 1 DMAX  = ',DMAX
007400       WRITE(6,'(1H ,A,F5.2)') ' 2 STEP  = ',STEP
007410       WRITE(6,'(1H ,A,F5.2)') ' 3 DMIN  = ',DMIN
007420 C
007430 C-----ASK FOR CHANGES.
007440   92  WRITE(6,'(/1H ,A)') 'ENTER NUMBER TO CHANGE, 0 TO CONTINUE'
007450 C
007460       READ(5,*,ERR=92,END=92) NUM
007470       IF (NUM.EQ.1) THEN
007480   94      WRITE(6,'(1H ,A)') 'ENTER DMAX'
007490           READ(5,*,ERR=94,END=94) DMAX
007500           IF (DMAX.LE.0.0.OR.DMAX.GT.1.0) THEN
007510               WRITE(6,'(1H ,A)') '  ENTER 0 < DMAX <= 1'
007520               GOTO 94
007530           ENDIF
007540         ELSE IF (NUM.EQ.2) THEN
007550   96      WRITE(6,'(1H ,A)') 'ENTER STEP'
007560           READ(5,*,ERR=96,END=96) STEP
007570           IF (STEP.LE.0.0.OR.STEP.GT.DMAX) THEN
007580               WRITE(6,'(1H ,A)') '  ENTER 0 < STEP <= DMAX'
007590               GOTO 96
007600           ENDIF
007610         ELSE IF (NUM.EQ.3) THEN
007620   98      WRITE(6,'(1H ,A)') 'ENTER DMIN'
007630           READ(5,*,ERR=98,END=98) DMIN
007640           IF (DMIN.LE.0.0.OR.DMIN.GT.DMAX) THEN
007650               WRITE(6,'(1H ,A)') '  ENTER 0 < DMIN <= DMAX'
007660               GOTO 98
007670           ENDIF
007680       ENDIF
007690       IF (NUM.NE.0) GOTO 90
007700 C
007710 C-----INITIALIZE OTHER PARAMETERS.
007720       NWT=1
007730       TT1(1)=2.08
007740       TT1(2)=2.16
007750       TT1(3)=2.08
007760       TT=2.03
007770 C
007780 C-----WRITE PARAMETERS.
007790   102 CONTINUE
007800       WRITE(6,'(/1H ,A)') '  OTHER PARAMETERS'
007810 C
007820       WRITE(6,'( 1H ,A,I3)') ' 1 NWT    = ',NWT
007830       DO 104 I=1,INY1
007840           WRITE(6,'(I3,A,I2,A,F6.2)') I+1,' TT1(',I,')= ',TT1(I)
007850   104 CONTINUE
007860       WRITE(6,'(1H ,I2,A,F6.2)') INY1+2,' TT     = ',TT
007870 C
007880 C-----ASK FOR CHANGES.
007890   105 WRITE(6,'(/1H ,A)') 'ENTER NUMBER TO CHANGE, 0 TO CONTINUE'
007900 C
007910       READ(5,*,ERR=105,END=105) NUM
007920       IF (NUM.EQ.1) THEN
```

```
007930  106      WRITE(6,'(1H ,A)') '  ENTER NWT'
007940           READ(5,*,ERR=106,END=106) NWT
007950           IF (NWT.LT.1.OR.NWT.GT.3) THEN
007960               WRITE(6,'(1H ,A)') '  ENTER 1, 2, OR 3'
007970               GOTO 106
007980           ENDIF
007990         ELSE IF (NUM.GT.1.AND.NUM.LE.INY1+1) THEN
008000  108      WRITE(6,'(1H ,A,I2,A)') 'ENTER TT1(',NUM-1,')'
008010           READ(5,*,ERR=108,END=108) TT1(NUM-1)
008020           IF (TT1(NUM-1).LE.0.0) THEN
008030               WRITE(6,'(1H ,A)') ' ENTER > 0.0'
008040               GOTO 108
008050           ENDIF
008060         ELSE IF (NUM.EQ.INY1+2) THEN
008070  200      WRITE(6,'(1H ,A)') 'ENTER TT'
008080           READ(5,*,ERR=200,END=200) TT
008090           IF (TT.LE.0.0) THEN
008100               WRITE(6,'(1H ,A)') '  ENTER > 0.0'
008110               GOTO 200
008120           ENDIF
008130       ENDIF
008140       IF (NUM.NE.0) GOTO 102
008150 C
008160 C
008170 C-----INITIALIZE OTHER PARAMETERS.
008180       DFACL=0.998
008190       DFACU=1.002
008200       N2=N+1
008210       DETC1=0.001
008220       DETC2=10.**(6.)
008230       ZFAC=10.**(0.5)
008240       INX=2
008250       INX1=INX+1
008260 C
008270 C
008280 C-----WRITE TABLE OF VALUES.
008290       WRITE (8,'(A)') '    TABLE OF INITIAL VALUES'
008300 C
008310       WRITE(8,'(//1H ,A)') 'PARAMETERS DESCRIBING SAMPLE'
008320       WRITE(8,'(/A,I2)') ' ISTART = ',ISTART-1
008330       WRITE(8,'(A,I2)') '      N = ',N-1
008340       WRITE(8,'(A,I2)') '    INY = ',INY
008350       WRITE(8,'(A)') ' IYEAR ARRAY'
008360       WRITE(8,'(15(2X,I2))') (IYEAR(I),I=1,INY)
008370 C
008380       WRITE(8,'(//A)')' PARAMETERS USED FOR SETTING INYF AND RHO'
008390       WRITE(8,'(/A,F6.3)') ' FACK1 = ',FACK1
008400       WRITE(8,'(A)') ' FACK ARRAY'
008410       WRITE(8,'(15(2X,F6.3)/)') (FACK(I),I=1,INY1)
008420       WRITE(8,'(/A,I1)') ' NTRY = ',NTRY
008430 C
008440       WRITE(8,'(//A)') ' PARAMETERS TO CONTROL NUMBER OF ITERATIONS'
008450       WRITE(8,'(/A)') ' IN2 ARRAY'
008460       WRITE(8,'(15(2X,I5)/)') (IN2(I),I=1,INY1)
008470       WRITE(8,'(/A)') ' IN3 ARRAY'
008480       WRITE(8,'(15(2X,I5)/)') (IN3(I),I=1,INY1)
008490       WRITE(8,'(/A,I5)')' KN1 = ',KN1
008500 C
008510       WRITE(8,'(//A)') ' PARAMETERS AFFECTING PRECISION OF SOLN'
008520       WRITE(8,'(/A)') ' FACK4 ARRAY'
008530       WRITE(8,'(15(1X,F8.7)/)') (FACK4(I),I=1,INY1)
008540       WRITE(8,'(/A,F5.2)') ' FCEPT = ', FCEPT
008550 C
008560       WRITE(8,'(//A)') ' VALUES USED FOR CONTROLLING D VALUE'
008570       WRITE(8,'(/A,F5.2)') ' DMAX  = ',DMAX
008580       WRITE(8,'(A,F5.2)')  ' STEP  = ',STEP
008590       WRITE(8,'(A,F5.2)')  ' DMIN  = ',DMIN
008600 C
008610       WRITE(8,'(//A)') ' OTHER PARAMETERS'
008620       WRITE(8,'(/A,I2)') ' NWT = ',NWT
008630       WRITE(8,'(A)') ' TT1 ARRAY'
008640       WRITE(8,'(15(2X,F5.2)/)') (TT1(I),I=1,INY1)
```

```
008650       WRITE(8,'(/A,F5.2)') ' TT = ',TT
008660 C
008670       WRITE(8,'(//A//)') ' END TABLE OF INITIAL VALUES'
008680 C
008690       RETURN
008700       END
008710 C
008720 C*******************************************************************************
008730 C
008740       FUNCTION FNX1(J,X,JW)
008750 CC
008760 CC----DESCRIPTION:  WEIGHTED SUM OF 'X(J,.)' USING WEIGHTS 'W1(JW,.)'.
008770 CC
008780 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
008790 CC                       WYCOVE SYSTEMS LIMITED,
008800 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
008810 CC                       RESEARCH BRANCH, DFO, HALIFAX.
008820 CC
008830 CC----DATE:  MARCH 1984.
008840 CC
008850 CC----REVISION HISTORY:  NONE.
008860 CC
008870 CC----FORMAL PARAMETERS:
008880 CC
008890 CC    PARAMETER  I/O  DESCRIPTION
008900 CC    -------------------------
008910 CC    J           I   INDEX INTO 'X(,)' MATRIX.
008920 CC    X           I   MATRIX WITH VALUES TO BE SUMMED.
008930 CC    JW          I   INDEX INTO 'W1(,)' MATRIX (SAMPLE NUMBER).
008940 CC
008950 CC----COMMON BLOCKS:
008960 CC
008970 CC    COMMON /AA/
008980 CC
008990 CC    VARIABLE  DESCRIPTION
009000 CC    ------------------
009010 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
009020 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
009030 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
009040 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
009050 CC    N2        N + 1.
009060 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
009070 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
009080 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
009090 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
009100 CC
009110 CC    COMMON /FNX/
009120 CC
009130 CC    VARIABLE  DESCRIPTION
009140 CC    ------------------
009150 CC    INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
009160 CC    INX1      INX + 1.
009170 CC
009180       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
009190       COMMON /FNX/ INX,INX1
009200 C
009210       DIMENSION X(INX1,30)
009220 C
009230       FNX1=0.
009240       DO 700 I=ISTART,N
009250           IF(W1(JW,I).LE.0.)GOTO 700
009260           FNX1=FNX1+X(J,I)*W1(JW,I)
009270   700 CONTINUE
009280       RETURN
009290       END
009300 C
009310 C*******************************************************************************
009320 C
009330       FUNCTION FNY1(J,Y)
009340 CC
009350 CC----DESCRIPTION:  WEIGHTED SUM OF 'Y(J,.)' USING WEIGHTS 'W1(J,.)'.
009360 CC
```

```
009370 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
009380 CC                          WYCOVE SYSTEMS LIMITED,
009390 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
009400 CC                          RESEARCH BRANCH, DFO, HALIFAX.
009410 CC
009420 CC----DATE:  MARCH 1984.
009430 CC
009440 CC----REVISION HISTORY:  NONE.
009450 CC
009460 CC----FORMAL PARAMETERS:
009470 CC
009480 CC    PARAMETER  I/O  DESCRIPTION
009490 CC    -------------------------
009500 CC    J           I   INDEX INTO 'Y(,)' AND 'W1(,)' MATRICES (SAMPLE
009510 CC                    NUMBER).
009520 CC    Y           I   MATRIX WITH VALUES TO BE SUMMED.
009530 CC
009540 CC----COMMON BLOCKS:
009550 CC
009560 CC    COMMON /AA/
009570 CC
009580 CC    VARIABLE  DESCRIPTION
009590 CC    --------------------
009600 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
009610 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
009620 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
009630 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
009640 CC    N2        N + 1.
009650 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
009660 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
009670 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
009680 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
009690 CC
009700 CC    COMMON /FNY/
009710 CC
009720 CC    VARIABLE  DESCRIPTION
009730 CC    --------------------
009740 CC    INY       NUMBER OF SAMPLES.
009750 CC    INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
009760 CC
009770       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
009780       COMMON /FNY/ INY,INY1
009790 C
009800       DIMENSION Y(10,30)
009810 C
009820       FNY1=0.
009830       DO 700 I=ISTART,N
009840           IF(W1(J,I).LE.0.)GOTO 700
009850           FNY1=FNY1+Y(J,I)*W1(J,I)
009860   700 CONTINUE
009870       RETURN
009880       END
009890 C
009900 C*******************************************************************
009910 C
009920       FUNCTION FNAX(J,X,JW)
009930 CC
009940 CC----DESCRIPTION:  WEIGHTED SUM OF 'X(J,.)' USING WEIGHTS 'W1(JW,.)'
009950 CC                  DIVIDED BY SUM OF THE WEIGHTS.
009960 CC
009970 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
009980 CC                          WYCOVE SYSTEMS LIMITED,
009990 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
010000 CC                          RESEARCH BRANCH, DFO, HALIFAX.
010010 CC
010020 CC----DATE:  MARCH 1984.
010030 CC
010040 CC----REVISION HISTORY:  NONE.
010050 CC
010060 CC----FORMAL PARAMETERS:
010070 CC
010080 CC    PARAMETER  I/O  DESCRIPTION
```

```
010090 CC    -------------------------
010100 CC    J           I   INDEX INTO 'X(,)' MATRIX.
010110 CC    X           I   MATRIX WITH VALUES TO BE SUMMED.
010120 CC    JW          I   INDEX INTO 'W1(,)' MATRIX (SAMPLE NUMBER).
010130 CC
010140 CC----EXTERNALS:  FUNCTION FNX1.
010150 CC
010160 CC----COMMON BLOCKS:
010170 CC
010180 CC    COMMON /AA/
010190 CC
010200 CC    VARIABLE  DESCRIPTION
010210 CC    --------------------
010220 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
010230 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
010240 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
010250 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
010260 CC    N2        N + 1.
010270 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
010280 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
010290 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
010300 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
010310 CC
010320 CC    COMMON /FNX/
010330 CC
010340 CC    VARIABLE  DESCRIPTION
010350 CC    --------------------
010360 CC    INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
010370 CC    INX1      INX + 1.
010380 CC
010390       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
010400       COMMON /FNX/ INX,INX1
010410 C
010420       DIMENSION X(INX1,30)
010430 C
010440       SUMW=0.
010450       DO 700 I=ISTART,N
010460           SUMW=SUMW+W1(JW,I)
010470   700 CONTINUE
010480       FNAX=FNX1(J,X,JW)/SUMW
010490       RETURN
010500       END
010510 C
010520 C*******************************************************************************
010530 C
010540       FUNCTION FNAY(J,Y)
010550 CC
010560 CC----DESCRIPTION:  WEIGHTED SUM OF 'Y(J,.)' USING WEIGHTS 'W1(J,.)'
010570 CC                  DIVIDED BY SUM OF THE WEIGHTS.
010580 CC
010590 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
010600 CC                        WYCOVE SYSTEMS LIMITED,
010610 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
010620 CC                        RESEARCH BRANCH, DFO, HALIFAX.
010630 CC
010640 CC----DATE:  MARCH 1984.
010650 CC
010660 CC----REVISION HISTORY:  NONE.
010670 CC
010680 CC----FORMAL PARAMETERS:
010690 CC
010700 CC    PARAMETER  I/O  DESCRIPTION
010710 CC    -------------------------
010720 CC    J           I   INDEX INTO 'Y(,)' AND 'W1(,)' MATRICES (SAMPLE
010730 CC                    NUMBER).
010740 CC    Y           I   MATRIX WITH VALUES TO BE SUMMED.
010750 CC
010760 CC----EXTERNALS:  FUNCTION FNY1.
010770 CC
010780 CC----COMMON BLOCKS:
010790 CC
010800 CC    COMMON /AA/
```

```
010810 CC
010820 CC    VARIABLE  DESCRIPTION
010830 CC    --------------------
010840 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
010850 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
010860 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
010870 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
010880 CC    N2        N + 1.
010890 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
010900 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
010910 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
010920 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
010930 CC
010940 CC    COMMON /FNY/
010950 CC
010960 CC    VARIABLE  DESCRIPTION
010970 CC    --------------------
010980 CC    INY       NUMBER OF SAMPLES.
010990 CC    INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
011000 CC
011010       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
011020       COMMON /FNY/ INY,INY1
011030 C
011040       DIMENSION Y(10,30)
011050 C
011060       SUMW=0.
011070       DO 700 I=ISTART,N
011080           SUMW=SUMW+W1(J,I)
011090   700 CONTINUE
011100       FNAY=FNY1(J,Y)/SUMW
011110       RETURN
011120       END
011130 C
011140 C*********************************************.**************************
011150 C
011160       FUNCTION FNXY(JX,JY,X,Y)
011170 CC
011180 CC----DESCRIPTION:  WEIGHTED SUM OF PRODUCTS OF 'X(JX,.)' AND 'Y(JY,.)'
011190 CC                  USING WEIGHTS 'W1(JY,.)'.
011200 CC
011210 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
011220 CC                          WYCOVE SYSTEMS LIMITED,
011230 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
011240 CC                          RESEARCH BRANCH, DFO, HALIFAX.
011250 CC
011260 CC----DATE:  MARCH 1984.
011270 CC
011280 CC----REVISION HISTORY:  NONE.
011290 CC
011300 CC----FORMAL PARAMETERS:
011310 CC
011320 CC    PARAMETER  I/O  DESCRIPTION
011330 CC    -------------------------
011340 CC    JX          I   INDEX INTO 'X(,)'.
011350 CC    JY          I   INDEX INTO 'Y(,)' AND 'W1(,)' MATRICES (SAMPLE
011360 CC                    NUMBER).
011370 CC    X           I   MATRIX WITH VALUES TO BE SUMMED.
011380 CC    Y           I   MATRIX WITH VALUES TO BE SUMMED.
011390 CC
011400 CC----COMMON BLOCKS:
011410 CC
011420 CC    COMMON /AA/
011430 CC
011440 CC    VARIABLE  DESCRIPTION
011450 CC    --------------------
011460 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
011470 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
011480 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
011490 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
011500 CC    N2        N + 1.
011510 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
011520 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
```

```
011530 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
011540 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
011550 CC
011560 CC    COMMON /FNX/
011570 CC
011580 CC    VARIABLE  DESCRIPTION
011590 CC    --------------------
011600 CC    INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
011610 CC    INX1      INX + 1.
011620 CC
011630 CC    COMMON /FNY/
011640 CC
011650 CC    VARIABLE  DESCRIPTION
011660 CC    --------------------
011670 CC    INY       NUMBER OF SAMPLES.
011680 CC    INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
011690 CC
011700       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
011710       COMMON /FNX/ INX,INX1
011720       COMMON /FNY/ INY,INY1
011730 C
011740       DIMENSION X(INX1,30),Y(10,30)
011750 C
011760       FNXY=0.
011770       DO 700 I=ISTART,N
011780           IF(W1(JY,I).LE.0.)GOTO 700
011790           FNXY=FNXY+W1(JY,I)*X(JX,I)*Y(JY,I)
011800   700 CONTINUE
011810       RETURN
011820       END
011830 C
011840 C*****************************************************************************
011850 C
011860       FUNCTION FNX2(J,J1,X,JW)
011870 CC
011880 CC----DESCRIPTION:  WEIGHTED SUM OF PRODUCTS OF 'X(J,.)' AND 'X(J1,)'
011890 CC                  USING WEIGHTS 'W1(JW,.)'.
011900 CC
011910 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
011920 CC                        WYCOVE SYSTEMS LIMITED,
011930 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
011940 CC                        RESEARCH BRANCH, DFO, HALIFAX.
011950 CC
011960 CC----DATE:  MARCH 1984.
011970 CC
011980 CC----REVISION HISTORY:  NONE.
011990 CC
012000 CC----FORMAL PARAMETERS:
012010 CC
012020 CC    PARAMETER  I/O  DESCRIPTION
012030 CC    -------------------------
012040 CC    J           I   INDEX INTO 'X(,)' MATRIX.
012050 CC    J1          I   INDEX INTO 'X(,)' MATRIX.
012060 CC    X           I   MATRIX WITH VALUES TO BE SUMMED.
012070 CC    JW          I   INDEX INTO 'W1(,)' MATRIX (SAMPLE NUMBER).
012080 CC
012090 CC----COMMON BLOCKS:
012100 CC
012110 CC    COMMON /AA/
012120 CC
012130 CC    VARIABLE  DESCRIPTION
012140 CC    --------------------
012150 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
012160 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
012170 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
012180 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
012190 CC    N2        N + 1.
012200 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
012210 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
012220 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
012230 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
012240 CC
```

```
012250 CC    COMMON /FNX/
012260 CC
012270 CC    VARIABLE  DESCRIPTION
012280 CC    --------------------
012290 CC    INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
012300 CC    INX1      INX + 1.
012310 CC
012320       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
012330       COMMON /FNX/ INX,INX1
012340 C
012350       DIMENSION X(INX1,30)
012360 C
012370       FNX2=0.
012380       DO 700 I=ISTART,N
012390           IF(W1(JW,I).LE.0.)GOTO 700
012400           FNX2=FNX2+W1(JW,I)*X(J,I)*X(J1,I)
012410   700 CONTINUE
012420       RETURN
012430       END
012440 C
012450 C*****************************************************************************
012460 C
012470       FUNCTION FNY2(J,Y)
012480 CC
012490 CC----DESCRIPTION:  WEIGHTED SUM OF SQUARES OF 'Y(J,.)' USING
012500 CC                  WEIGHTS 'W1(J,.)'.
012510 CC
012520 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
012530 CC                       WYCOVE SYSTEMS LIMITED,
012540 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
012550 CC                       RESEARCH BRANCH, DFO, HALIFAX.
012560 CC
012570 CC----DATE:  MARCH 1984.
012580 CC
012590 CC----REVISION HISTORY:  NONE.
012600 CC
012610 CC----FORMAL PARAMETERS:
012620 CC
012630 CC    PARAMETER  I/O  DESCRIPTION
012640 CC    -------------------------
012650 CC    J           I   INDEX INTO 'Y(,)' AND 'W1(,)' MATRICES (SAMPLE
012660 CC                    NUMBER).
012670 CC    Y           I   MATRIX WITH VALUES TO BE SUMMED.
012680 CC
012690 CC----COMMON BLOCKS:
012700 CC
012710 CC    COMMON /AA/
012720 CC
012730 CC    VARIABLE  DESCRIPTION
012740 CC    --------------------
012750 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
012760 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
012770 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
012780 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
012790 CC    N2        N + 1.
012800 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
012810 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
012820 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
012830 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
012840 CC
012850 CC    COMMON /FNY/
012860 CC
012870 CC    VARIABLE  DESCRIPTION
012880 CC    --------------------
012890 CC    INY       NUMBER OF SAMPLES.
012900 CC    INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
012910 CC
012920       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
012930       COMMON /FNY/ INY,INY1
012940 C
012950       DIMENSION Y(10,30)
012960 C
```

```
012970       FNY2=0.
012980       DO 700 I=ISTART,N
012990           IF(W1(J,I).LE.0.)GOTO 700
013000           FNY2=FNY2+W1(J,I)*Y(J,I)**2
013010   700 CONTINUE
013020       RETURN
013030       END
013040 C
013050 C*******************************************************************************
013060 C
013070       FUNCTION FNSSX(J,J1,X,JW)
013080 CC
013090 CC----DESCRIPTION:  WEIGHTED SUM OF PRODUCTS OF 'X(J,.)' AND 'X(J1,)'
013100 CC                  MINUS (PRODUCT OF WEIGHTED SUMS OF 'X(J,.)' AND
013110 CC                  'X(J1,.)' DIVIDED BY SUM OF WEIGHTS).
013120 CC
013130 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
013140 CC                         WYCOVE SYSTEMS LIMITED,
013150 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
013160 CC                         RESEARCH BRANCH, DFO, HALIFAX.
013170 CC
013180 CC----DATE:  MARCH 1984.
013190 CC
013200 CC----REVISION HISTORY:  NONE.
013210 CC
013220 CC----FORMAL PARAMETERS:
013230 CC
013240 CC    PARAMETER  I/O  DESCRIPTION
013250 CC    -------------------------
013260 CC    J           I   INDEX INTO 'X(,)' MATRIX.
013270 CC    J1          I   INDEX INTO 'X(,)' MATRIX.
013280 CC    X           I   MATRIX WITH VALUES TO BE SUMMED.
013290 CC    JW          I   INDEX INTO 'W1(,)' MATRIX (SAMPLE NUMBER).
013300 CC
013310 CC----EXTERNALS:  FUNCTIONS FNX1, FNX2.
013320 CC
013330 CC----COMMON BLOCKS:
013340 CC
013350 CC    COMMON /AA/
013360 CC
013370 CC    VARIABLE  DESCRIPTION
013380 CC    --------------------
013390 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
013400 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
013410 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
013420 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
013430 CC    N2        N + 1.
013440 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
013450 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
013460 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
013470 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
013480 CC
013490 CC    COMMON /FNX/
013500 CC
013510 CC    VARIABLE  DESCRIPTION
013520 CC    --------------------
013530 CC    INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
013540 CC    INX1      INX + 1.
013550 CC
013560       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
013570       COMMON /FNX/ INX,INX1
013580 C
013590       DIMENSION X(INX1,30)
013600 C
013610       SUMW=0.
013620       DO 700 I=ISTART,N
013630           SUMW=SUMW+W1(JW,I)
013640   700 CONTINUE
013650       FNSSX=FNX2(J,J1,X,JW)-FNX1(J,X,JW)*FNX1(J1,X,JW)/SUMW
013660       RETURN
013670       END
013680 C
```

```
013690 C*************************************************************************
013700 C
013710       FUNCTION FNSSY(J,Y)
013720 CC
013730 CC----DESCRIPTION:  WEIGHTED SUM OF SQUARES OF 'Y(J,.)' MINUS (SQUARE OF
013740 CC                  WEIGHTED SUM OF 'Y(J,.)' DIVIDED BY SUM OF WEIGHTS).
013750 CC
013760 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
013770 CC                          WYCOVE SYSTEMS LIMITED,
013780 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
013790 CC                          RESEARCH BRANCH, DFO, HALIFAX.
013800 CC
013810 CC----DATE:  MARCH 1984.
013820 CC
013830 CC----REVISION HISTORY:  NONE.
013840 CC
013850 CC----FORMAL PARAMETERS:
013860 CC
013870 CC    PARAMETER  I/O  DESCRIPTION
013880 CC    -------------------------
013890 CC    J           I   INDEX INTO 'Y(,)' AND 'W1(,)' MATRICES (SAMPLE
013900 CC                    NUMBER).
013910 CC    Y           I   MATRIX WITH VALUES TO BE SUMMED.
013920 CC
013930 CC----EXTERNALS:  FUNCTIONS FNY1, FNY2.
013940 CC
013950 CC----COMMON BLOCKS:
013960 CC
013970 CC    COMMON /AA/
013980 CC
013990 CC    VARIABLE  DESCRIPTION
014000 CC    --------------------
014010 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
014020 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
014030 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
014040 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
014050 CC    N2        N + 1.
014060 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
014070 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
014080 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
014090 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
014100 CC
014110 CC    COMMON /FNY/
014120 CC
014130 CC    VARIABLE  DESCRIPTION
014140 CC    --------------------
014150 CC    INY       NUMBER OF SAMPLES.
014160 CC    INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
014170 CC
014180       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
014190       COMMON /FNY/ INY,INY1
014200 C
014210       DIMENSION Y(10,30)
014220 C
014230       SUMW=0.
014240       DO 700 I=ISTART,N
014250          SUMW=SUMW+W1(J,I)
014260   700 CONTINUE
014270       FNSSY=FNY2(J,Y)-FNY1(J,Y)**2/SUMW
014280       RETURN
014290       END
014300 C
014310 C*************************************************************************
014320 C
014330       FUNCTION FNSP(JX,JY,X,Y)
014340 CC
014350 CC----DESCRIPTION:  WEIGHTED SUM OF PRODUCTS OF 'X(JX,.)' AND 'Y(JY,)'
014360 CC                  MINUS (PRODUCT OF WEIGHTED SUMS OF 'X(JX,.)' AND
014370 CC                  'Y(JY,.)' DIVIDED BY SUM OF WEIGHTS).
014380 CC
014390 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
014400 CC                          WYCOVE SYSTEMS LIMITED,
```

```
014410 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
014420 CC                          RESEARCH BRANCH, DFO, HALIFAX.
014430 CC
014440 CC----DATE:  MARCH 1984.
014450 CC
014460 CC----REVISION HISTORY:  NONE.
014470 CC
014480 CC----FORMAL PARAMETERS:
014490 CC
014500 CC    PARAMETER  I/O  DESCRIPTION
014510 CC    -------------------------
014520 CC    JX          I   INDEX INTO 'X(,)' MATRIX.
014530 CC    JY          I   INDEX INTO 'Y(,)' AND 'W1(,)' MATRICES (SAMPLE
014540 CC                    NUMBER).
014550 CC    X           I   MATRIX WITH VALUES TO BE SUMMED.
014560 CC    Y           I   MATRIX WITH VALUES TO BE SUMMED.
014570 CC
014580 CC----EXTERNALS:  FUNCTIONS FNX1, FNY1, FNXY.
014590 CC
014600 CC----COMMON BLOCKS:
014610 CC
014620 CC    COMMON /AA/
014630 CC
014640 CC    VARIABLE  DESCRIPTION
014650 CC    --------------------
014660 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
014670 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
014680 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
014690 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
014700 CC    N2        N + 1.
014710 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
014720 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
014730 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
014740 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
014750 CC
014760 CC    COMMON /FNX/
014770 CC
014780 CC    VARIABLE  DESCRIPTION
014790 CC    --------------------
014800 CC    INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
014810 CC    INX1      INX + 1.
014820 CC
014830 CC    COMMON /FNY/
014840 CC
014850 CC    VARIABLE  DESCRIPTION
014860 CC    --------------------
014870 CC    INY       NUMBER OF SAMPLES.
014880 CC    INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
014890 CC
014900       COMMON /AA/  ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
014910       COMMON /FNX/ INX,INX1
014920       COMMON /FNY/ INY,INY1
014930 C
014940       DIMENSION X(INX1,30),Y(10,30)
014950 C
014960       SUMW=0.
014970       DO 700 I=ISTART,N
014980           SUMW=SUMW+W1(JY,I)
014990   700 CONTINUE
015000       FNSP=FNXY(JX,JY,X,Y)-FNX1(JX,X,JY)*FNY1(JY,Y)/SUMW
015010       RETURN
015020       END
015030 C
015040 C*******************************************************************************
015050 C
015060       SUBROUTINE MISRA1
015070 CC
015080 CC----DESCRIPTION:  USES METHOD DESCRIBED IN MISRA AND IN BLISS TO
015090 CC                  FIT A VON BERTALANFFY GROWTH CURVE TO EACH SAMPLE.
015100 CC                  IF 'D' VALUE FOR ALL SAMPLES IS EQUAL, PASSES ON
015110 CC                  TO 'MISRA2' TO FIT A CURVE TO THE COMBINED SAMPLE.
015120 CC
```

```
015130 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
015140 CC                          WYCOVE SYSTEMS LIMITED,
015150 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
015160 CC                          RESEARCH BRANCH, DFO, HALIFAX.
015170 CC
015180 CC----DATE:  MARCH 1984.
015190 CC
015200 CC----REVISION HISTORY:  NONE.
015210 CC
015220 CC----FILES:
015230 CC
015240 CC     UNIT   I/O   DESCRIPTION
015250 CC     ----------------------
015260 CC       8     O    RESULTS AND MESSAGES.
015270 CC
015280 CC----FORMAL PARAMETERS:  NONE.
015290 CC
015300 CC----EXTERNALS:
015310 CC
015320 CC     NAME      DESCRIPTION
015330 CC     -------------------
015340 CC     MISRA2    GENERATES FIT FOR COMBINED SAMPLE.
015350 CC     MISRA5    ESTIMATES INITIAL VALUE OF 'YINF'.
015360 CC     MISRA6    GENERATES 'K' AND 'T0' PARAMETERS.
015370 CC     MISRA7    GENERATES PARAMETERS USING ALLEN'S METHOD.
015380 CC     TRY1      ESTIMATES INITIAL VALUE OF 'RHO' USING WALFORD LINE.
015390 CC     TRY2      ESTIMATES INITIAL VALUE OF 'RHO' USING FIRST AND MIDDLE
015400 CC               AGE GROUPS.
015410 CC     TRY3      ESTIMATES INITIAL VALUE OF 'RHO' USING LOGARITHMIC FIT
015420 CC               (ONLY ONE OF 'TRY1', 'TRY2', OR 'TRY3' USED).
015430 CC     SMPY      MULTIPLIES A MATRIX BY A SCALAR.
015440 CC     GMPRD     MATRIX PRODUCT.
015450 CC     MINV      MATRIX INVERSE.
015460 CC     FUNCTIONS  FNX1, FNX2, FNY1, FNXY, FNAX, FNAY, FNSP, FNSSX, FNSSY.
015470 CC
015480 CC----COMMON BLOCKS:
015490 CC
015500 CC     COMMON /AA/
015510 CC
015520 CC     VARIABLE  DESCRIPTION
015530 CC     --------------------
015540 CC     ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
015550 CC               THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
015560 CC     N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
015570 CC               THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
015580 CC     N2        N + 1.
015590 CC     W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
015600 CC     W3(10,30) TEMPORARY COPY OF 'W1(,)'.
015610 CC     LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
015620 CC     LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
015630 CC
015640 CC     COMMON /ONER/
015650 CC
015660 CC     VARIABLE  DESCRIPTION
015670 CC     --------------------
015680 CC     IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
015690 CC               ESTIMATE OF 'RHO'.
015700 CC     TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
015710 CC     TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
015720 CC               DATA.
015730 CC     Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
015740 CC               AGE + 1.
015750 CC     DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
015760 CC     N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
015770 CC     SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
015780 CC     HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
015790 CC     IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
015800 CC               VALUE FOR 'B1/B'.
015810 CC     FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
015820 CC     IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
015830 CC               OF YEAR SAMPLE COLLECTED.
015840 CC     NWT       TYPE OF WEIGHTING:
```

```
015850 CC              1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
015860 CC              2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
015870 CC              3 - CONSTANT = 1.
015880 CC     NTRY     SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
015890 CC              IF 'NTRY'=1 USE 'TRY1', ETC.
015900 CC
015910 CC     COMMON /FNX/
015920 CC
015930 CC     VARIABLE  DESCRIPTION
015940 CC     --------------------
015950 CC     INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
015960 CC     INX1      INX + 1.
015970 CC
015980 CC     COMMON /FNY/
015990 CC
016000 CC     VARIABLE  DESCRIPTION
016010 CC     --------------------
016020 CC     INY       NUMBER OF SAMPLES.
016030 CC     INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
016040 CC
016050 CC     COMMON /AA1/
016060 CC
016070 CC     VARIABLE  DESCRIPTION
016080 CC     --------------------
016090 CC     RBERT1(10)   INITIAL ESTIMATE OF 'RHO' FROM 'TRY1' (2 OR 3).
016100 CC
016110 CC     COMMON /AA2/
016120 CC
016130 CC     VARIABLE  DESCRIPTION
016140 CC     --------------------
016150 CC     ITRY      CONTROLS HEADINGS IN 'TRY' ROUTINES.
016160 CC     AYINF     IN 'TRY2' AND 'TRY3', SET EQUAL TO NUMBER OF TIMES
016170 CC               ESTIMATE OF 'YINF' HAS TO BE MULTIPLIED BY 'FACK()'
016180 CC               TO GET A REASONABLE VALUE OF 'RHO'.
016190 CC
016200 CC     COMMON /AA3/
016210 CC
016220 CC     VARIABLE  DESCRIPTION
016230 CC     --------------------
016240 CC     YINF1     INITIAL ESTIMATE OF 'YINF' SET IN 'MISRA5'.
016250 CC
016260 CC     COMMON /AA4/
016270 CC
016280 CC     VARIABLE  DESCRIPTION
016290 CC     --------------------
016300 CC     RBERT2(10)   FINAL ESTIMATE OF 'RHO' FROM 'MISRA1'.
016310 CC
016320 CC     COMMON /AA10/
016330 CC
016340 CC     VARIABLE  DESCRIPTION
016350 CC     --------------------
016360 CC     FACK(10)  FACTOR FOR INCREASING ESTIMATE OF 'YINF' IN 'TRY2' OR
016370 CC               'TRY3'.
016380 CC     KN1       MAXIMUM NUMBER OF ITERATIONS OF LOOP THAT RE-ESTIMATES
016390 CC               'YINF' BY 'TRY' ROUTINES, THEN REDOES REGRESSION.
016400 CC
016410 CC     COMMON /AA11/
016420 CC
016430 CC     VARIABLE  DESCRIPTION
016440 CC     --------------------
016450 CC     DFAC(10)  'D' VALUE FOR SAMPLE.
016460 CC     DMAX      STARTING (MAXIMUM) VALUE FOR 'D'.
016470 CC     STEP      VALUE SUBTRACTED FROM 'D' EACH ITERATION.
016480 CC     DMIN      MINIMUM VALUE FOR 'D'.
016490 CC     Y4(10,30) TEMPORARY STORAGE FOR 'Y(,)'.
016500 CC
016510 CC     COMMON /AA15/
016520 CC
016530 CC     VARIABLE  DESCRIPTION
016540 CC     --------------------
016550 CC     FCEPT     ACCEPTANCE CONTROL ON 'YINF', MUST BE LESS THAN 'FCEPT'
016560 CC               TIMES REVISED ESTIMATE OF 'YINF' FROM 'TRY' ROUTINES.
```

```
016570 CC
016580 CC     COMMON /AA19/
016590 CC
016600 CC     VARIABLE  DESCRIPTION
016610 CC     --------------------
016620 CC     DETC1     MIN VALUE ACCEPTABLE FOR DETERMINANT IF INVERTING
016630 CC               MATRIX.
016640 CC     DETC2     MAX VALUE ACCEPTABLE FOR DETERMINANT IF INVERTING
016650 CC               MATRIX.
016660 CC     ZFAC      FACTOR FOR SCALING MATRIX IF DETERMINANT TOO BIG OR
016670 CC               SMALL.
016680 CC
016690 CC     COMMON /COM/
016700 CC
016710 CC     VARIABLE  DESCRIPTION
016720 CC     --------------------
016730 CC     SCASSE    VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
016740 CC               ANOVA.
016750 CC     ISCAE     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
016760 CC               ANOVA.
016770 CC     SCASS     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
016780 CC               ANOVA.
016790 CC     ISCA      VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
016800 CC               ANOVA.
016810 CC     SUMRSS    VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
016820 CC               ANOVA.
016830 CC     COMSS     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
016840 CC               ANOVA.
016850 CC     SR(2,10)  VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
016860 CC               ANOVA.
016870 CC     SR1(10)   VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
016880 CC               ANOVA.
016890 CC
016900 CC     COMMON /CCM/
016910 CC
016920 CC     VARIABLE  DESCRIPTION
016930 CC     --------------------
016940 CC     PARM(10,3)   FIT PARAMETERS FOR EACH SAMPLE:  'YINF', 'B', 'RHO'.
016950 CC     VPARM(10,3)  VARIANCE ESTIMATES FOR FIT PARAMETERS.
016960 CC
016970 CC----STATEMENT FUNCTIONS:
016980 CC
016990 CC     PROTOTYPE  DESCRIPTION
017000 CC     ---------------------
017010 CC     SN1(T)      'RHO**T' (THIS IS 'Z' IN MISRA PAPER).
017020 CC     SN2(T)      'T*RHO**(T-1)' (THIS IS 'Z1' IN MISRA PAPER).
017030 CC     SN3(T)      PREDICTED LENGTH.
017040 CC
017050       DOUBLE PRECISION ZFAC1,ZFAC2,CRIT,DCC1,BBMAT(3,1),
017060      >                 CC1(3,3),XY1(3,1)
017070       SN1(K)=RBERT**K
017080       SN2(K)=K*SN1(K)/RBERT
017090       SN3(K)=ACEPT+BMAT1*SN1(K)
017100 C
017110       COMMON /AA/   ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
017120       COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
017130      >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
017140       COMMON /FNX/  INX,INX1
017150       COMMON /FNY/  INY,INY1
017160       COMMON /AA1/  RBERT1(10)
017170       COMMON /AA2/  ITRY,AYINF
017180       COMMON /AA3/  YINF1
017190       COMMON /AA4/  RBERT2(10)
017200       COMMON /AA10/ FACK(10),KN1
017210       COMMON /AA11/ DFAC(10),DMAX,STEP,DMIN,Y4(10,30)
017220       COMMON /AA15/ FCEPT
017230       COMMON /AA19/ DETC1,DETC2,ZFAC
017240       COMMON /COM/  SCASSE,ISCAE,SCASS,ISCA,SUMRSS,COMSS,SR(2,10),
017250      >              SR1(10)
017260       COMMON /CMM/  PARM(10,3),VPARM(10,3)
017270 C
017280 C
```

```
017290       DIMENSION LW(3),MW(3)
017300       DIMENSION X(3,30),VEXPY(10,30),SEXPY(10,30),VPARMM(10)
017310       DIMENSION Y5(30),ECC1(3,3),EBMAT(2,1),EVEXPY(30),ESEXPY(30)
017320       DIMENSION BMAT(2,1),SSR(2),SBMAT(2,1),CONFL(2,1),CONFU(2,1),FR(2)
017330 C
017340       SUMRS=0.
017350   600 FORMAT('1')
017360   601 FORMAT(/)
017370   602 FORMAT(//)
017380       DO 746 I=1,N2
017390           DO 746 J=1,INX
017400               X(J,I)=0.
017410   746 CONTINUE
017420       SCASS=0.
017430       ISCA=0
017440       SUMRSS=0.
017450 C
017460 C-----SOLVE FOR EACH SAMPLE.
017470 C
017480       DO 701 I=1,INY
017490           WRITE(8,600)
017500           WRITE(8,611)I
017510   611     FORMAT(25X,'FOR  SAMPLE ',I2,2X,'(EQUATION 1)'//)
017520           SDM1=0.
017530           EDFAC=0.
017540           ISTEP=0
017550           ZFAC1=ZFAC
017560           SW=0.
017570           DO 730 I1=ISTART,N
017580               SW=SW+W1(I,I1)
017590   730     CONTINUE
017600           DFAC(I)=DMAX
017610           INN2=IN2(I)
017620           INN3=IN3(I)
017630           ITRY=0
017640 C
017650 C-----LOOP THROUGH VALUES FOR 'D', REPRESENTED HERE BY 'DFAC()'.  IT
017660 C     STARTS WITH A MAXIMUM VALUE OF 'DMAX', IS REDUCED EACH LOOP
017670 C     BY 'STEP', AND HAS A MINIMUM VALUE OF 'DMIN'.  ALL THESE
017680 C     PARAMETERS CAN BE SET BY USER.
017690 C
017700           DO 1000 ID=1,1000
017710 C
017720 C-----SUBROUTINE 'MISRA5' WILL ESTIMATE 'YINF'.  THIS VALUE IS RAISED
017730 C     IF NO SOLUTION CAN BE FOUND AT GIVEN VALUE.  'AYINF' WILL CONTAIN
017740 C     THE RATIO INCREASE IN 'YINF'.  'Y(,)' IS RESET TO THE ORIGINAL
017750 C     VALUE RAISED TO POWER 'DFAC()'.  SUBROUTINE 'TRY1' (2 OR 3) IS
017760 C     USED TO GET FIRST ESTIMATE OF 'RHO' FOR A GIVEN VALUE OF 'YINF'.
017770 C     'RHO' ESTIMATE IS IN 'RBERT'.  'RBERT1()' IS ORIGINAL ESTIMATE,
017780 C     'RBERT2()' IS BEST ESTIMATE.
017790 C
017800               AYINF=1.
017810               DO 799 J=ISTART,N2
017820                   Y(I,J)=Y4(I,J)
017830                   IF(Y(I,J).GT.0.0001) THEN
017840                       Y(I,J)=Y(I,J)**DFAC(I)
017850                   ENDIF
017860   799         CONTINUE
017870               CALL MISRA5(I)
017880 C
017890 C-----LOOP ON INCREMENTED 'YINF' VALUE.  RE-ESTIMATE 'RHO' AND TRY
017900 C     AGAIN.  ONLY LOOP 'KN1' TIMES.  'KN1' CAN BE SET BY USER FOR
017910 C     'TRY2' AND 'TRY3'.  FOR 'TRY1' IT IS HELD TO 1.
017920 C
017930               DO 990 IT=1,KN1
017940                   IF (NTRY.EQ.1) THEN
017950                       CALL TRY1(I)
017960                     ELSE IF (NTRY.EQ.2) THEN
017970                       CALL TRY2(I)
017980                     ELSE
017990                       CALL TRY3(I)
018000                   ENDIF
```

```
018010                    RBERT=RBERT1(I)
018020                    ACEPT1=YINF1*AYINF
018030                    ACEPT1=FCEPT*ACEPT1
018040 C
018050 C-----LOOP TO CONTROL THE PRECISION OF THE TEST ON 'B1/B'.
018060 C     TEST VALUE IS SET TO 'FACK4()' AT FIRST AND IS
018070 C     INCREMENTED BY 'FACK4()' EACH ITERATION.  MAX NUMBER
018080 C     OF ITERATIONS COMES FROM 'IN3()'.  'FACK4()' AND
018090 C     'IN3()' CAN BE SET BY USER.
018100 C
018110                    A1A=0.
018120                    DO 980 I1=1,INN3
018130                        A1A=A1A+FACK4(I)
018140 C
018150 C-----LOOP TO IMPROVE ESTIMATE OF 'RHO' BY ADDING CALCULATED
018160 C     VALUE OF 'B1/B'.  CUT OFF IF LESS THAN VALUE SET IN
018170 C     980 LOOP.  MAX NUMBER OF ITERATIONS COMES FROM 'IN2()',
018180 C     WHICH CAN BE SET BY USER.
018190 C
018200                        DO 970 IIN2=1,INN2
018210 C
018220 C-----'X(,)' CONTAINS 'Z' AND 'Z1'.
018230 C
018240                            DO 753 I2=ISTART,N
018250                                I3=I2-1
018260                                X(1,I2)=SN1(I3)
018270                                X(2,I2)=SN2(I3)
018280  753                       CONTINUE
018290                            DO 4156 J=1,INX
018300                                XY1(J+1,1)=FNSP(J,I,X,Y)
018310  4156                      CONTINUE
018320                            DO 4157 II1=2,INX1
018330                             DO 4157 J=2,INX1
018340                                CC1(II1,J)=FNSSX(II1-1,J-1,X,I)
018350  4157                      CONTINUE
018360                            A1BB=CC1(3,3)*XY1(2,1)-CC1(2,3)*XY1(3,1)
018370                            A1B=CC1(2,2)*XY1(3,1)-CC1(2,3)*XY1(2,1)
018380 C
018390 C-----EQUATION 16.35, P.149 OF BLISS.  THIS IS 'B1/B'.
018400 C
018410                            A1B=A1B/A1BB
018420                            A1BB=ABS(A1B)
018430                            IF(A1BB.LE.A1A) THEN
018440 C
018450 C-----IF 'B1/B' IS SMALL, THEN WORK OUT THE PARAMETERS OF THE EQUATION
018460 C     AND CHECK IF FIT IS OKAY.  SEE BLISS P.147-149.
018470 C
018480                                DO 731 J=1,INX
018490                                    XY1(J+1,1)=FNXY(J,I,X,Y)
018500  731                           CONTINUE
018510 C
018520 C-----THIS LOOP HANDLES INVERSE OF COEF. MATRIX OF NORMAL EQUATIONS.
018530 C     IF DETERMINANT IS TOO EXTREME, THEN MATRIX IS SCALED.
018540 C     'SMPY' IS SCALAR MULT. 1ST TERM * 2ND, PUT IN THIRD.
018550 C     'GMPRD' IS MATRIX PROD. 1ST TERM * 2ND, PUT IN THIRD.
018560 C     'MINV' IS INVERSE OF 1ST TERM, RETURN IN FIRST, DETERMINANT
018570 C     IN SECOND.
018580 C
018590                                DO 4036 IINV=1,100
018600                                    CC1(1,1)=SW
018610                                    DO 732 II1=2,INX1
018620                                        CC1(1,II1)=FNX1(II1-1,X,I)
018630                                        CC1(II1,1)=CC1(1,II1)
018640                                      DO 732 J=2,INX1
018650                                          CC1(II1,J)=FNX2(II1-1,J-1,X,I)
018660  732                               CONTINUE
018670                                    XY1(1,1)=FNY1(I,Y)
018680                                    CALL SMPY(CC1,ZFAC1,CC1,INX1,INX1,0)
018690                                    DCC1=0.
018700                                    CALL MINV(CC1,INX1,DCC1,LW,MW)
018710                                    CALL SMPY(CC1,ZFAC1,CC1,INX1,INX1,0)
018720                                    IF(DCC1.LE.DETC1) THEN
```

```
018730                                      ZFAC1=ZFAC1*2.14
018740                                    ELSE IF(DCC1.GE.DETC2) THEN
018750                                      ZFAC1=ZFAC1*0.47
018760                                    ELSE
018770                                      GOTO 4035
018780                                    ENDIF
018790  4036                            CONTINUE
018800  4035                            CALL GMPRD(CC1,XY1,BBMAT,INX1,INX1,1)
018810 C
018820 C-----'ACEPT2' IS CALCULATED VALUE OF 'YINF'.  IF THIS VALUE IS LESS
018830 C     THAN ZERO OR IS GREATER THAN 'ACEPT1', THIS SOLUTION IS REJECTED.
018840 C     'ACEPT1' IS 'YINF' AFTER 'TRY1' (2 OR 3) * 'FCEPT', A FACTOR
018850 C     THAT CAN BE SET BY THE USER.  SOLUTION IS ALSO REJECTED IF THE
018860 C     ESTIMATE OF 'BETA' ('BBMAT(2,2)') IS POSITIVE, OR IF 'B1'
018870 C     ('BBMAT(3,1)') IS MORE THAN 'TT1()' * (1 STANDARD ERROR OF 'B1')
018880 C     FROM ZERO.  'TT1()' CAN BE SET BY THE USER.
018890 C
018900                                  ACEPT2=FNAY(I,Y)-BBMAT(2,1)*FNAX(1,X,I)
018910                                  IF(ACEPT2.LE.0.) GOTO 969
018920                                  IF(ACEPT2.GT.ACEPT1) GOTO 969
018930                                  IF(BBMAT(2,1).GT.0.) GOTO 969
018940                                  RSSSA=FNSSY(I,Y)-BBMAT(2,1)*FNSP(1,I,X,Y)
018950      >                                 -BBMAT(3,1)*FNSP(2,I,X,Y)
018960                                  RSMSA=N-ISTART-INX-LNY(I)
018970                                  RSMSA=RSSSA/RSMSA
018980                                  CRIT=DSQRT((CC1(3,3)/HARN1(I))*RSMSA)
018990                                  CRIT=TT1(I)*CRIT
019000                                 IF(DABS(BBMAT(3,1)).GT.DABS(CRIT))GOTO 969
019010                                  GOTO 999
019020                                ELSE
019030  969                             RBERT=RBERT+A1B
019040                                  IF (RBERT.GE.1.0) GOTO 990
019050                                ENDIF
019060  970                         CONTINUE
019070  980                     CONTINUE
019080  990                 CONTINUE
019090 C
019100 C-----IF WE FINISH THE 990 LOOP WITHOUT JUMPING TO 999, THEN THERE IS
019110 C     NO ACCEPTABLE SOLUTION FOR THIS VALUE OF 'DFAC()', SO GOTO 1000.
019120 C
019130                  DFAC(I)=DFAC(I) - STEP
019140                  IF (DFAC(I).LT.DMIN.OR.DFAC(I).LE.0) GOTO 1001
019150                  GOTO 1000
019160 C
019170 C-----JUMPS TO 999 IF SOLUTION IS ACCEPTABLE.  NOW ESTIMATE PARAMETERS
019180 C     AND SEE IF THIS IS THE BEST FIT SO FAR.  'ACEPT' IS ESTIMATE OF
019190 C     'YINF'.  'REGSS' IS SUM OF SQUARES ATTRIBUTABLE TO REGRESSION,
019200 C     16.44 BLISS.  'RESMS' IS ERROR VARIANCE, 16.45 BLISS.  'BMAT1' IS
019210 C     ESTIMATE OF 'RHO'.  'VEXPY(,)' IS THE VARIANCE OF THE PREDICTED
019220 C     RESPONSE, 16.50 BLISS.  'SDM' IS THE SUM OF (ABSOLUTE VALUE OF
019230 C     RESIDUALS / S.E. OF THE PREDICTED VALUE).  BEST SOLUTION HAS
019240 C     MINIMUM 'SDM'.
019250 C
019260  999             REGSS=0.
019270                  ACEPT=FNAY(I,Y)-BBMAT(2,1)*FNAX(1,X,I)
019280                  SDM=0.
019290                  RESSS=0.
019300                  DO 5443 J=1,INX
019310                      BMAT(J,1)=BBMAT(J+1,1)
019320                      REGSS=REGSS+BMAT(J,1)*FNSP(J,I,X,Y)
019330  5443            CONTINUE
019340                  BMAT1=BMAT(1,1)
019350                  RESSS=FNSSY(I,Y)-REGSS
019360                  AR=REGSS/FNSSY(I,Y)
019370                  IRESSS=N-ISTART-INX-LNY(I)
019380                  RESMS=RESSS/IRESSS
019390                  KJ2=0
019400                  DO 5427 J2=ISTART,N
019410                      IF(W1(I,J2).EQ.0.0) THEN
019420                        KJ2=KJ2+1
019430                      ENDIF
019440  5427            CONTINUE
```

```
019450                KJ2=N-ISTART+1-KJ2
019460                VPARMM(I)=1./FLOAT(KJ2)
019470                DO 3048 J=1,INX
019480                  DO 3048 J1=1,INX
019490                    VPARMM(I)=(VPARMM(I)+FNAX(J,X,I)*FNAX(J1,X,I)
019500       >                        *CC1(J+1,J1+1))
019510   3048         CONTINUE
019520                VPARMM(I)=VPARMM(I)/HARN1(I)
019530                DO 784 J2=ISTART,N
019540                    VEXPY(I,J2)=X(1,J2)*(2.*CC1(1,2)+X(1,J2)*CC1(2,2)
019550       >                +2.*X(2,J2)*CC1(2,3))
019560                    VEXPY(I,J2)=VEXPY(I,J2)+X(2,J2)*(2.*CC1(1,3)
019570       >                +X(2,J2)*CC1(3,3))
019580                    VEXPY(I,J2)=(VPARMM(I)+VEXPY(I,J2)/HARN1(I))*RESMS
019590                    IF(VEXPY(I,J2).GE.0.) THEN
019600                        SEXPY(I,J2)=SQRT(VEXPY(I,J2))
019610                    ENDIF
019620    784           CONTINUE
019630          SDM=1-AR
019640 C
019650 C-----IF 'SDM' IS ZERO, WE CANNOT DO BETTER.
019660 C
019670                IF(SDM.LE.0.)GOTO 5434
019680 C
019690 C-----IF 'SDM' LESS THAN 'EMIN' (BEST 'SDM' SO FAR), WE WANT TO
019700 C     SAVE THE SOLUTION.  'ISTEP' COUNTS SAVED SOLUTIONS.
019710 C
019720                IF(ISTEP.LT.1.OR.SDM.LT.EMIN) THEN
019730                    ISTEP=ISTEP+1
019740                    EDFAC=DFAC(I)
019750                    EA1A=A1A
019760                    EVPARM=VPARMM(I)
019770                    EMIN=SDM
019780                    EACEPT=ACEPT
019790                    ERBERT=RBERT
019800                    DO 5436 J=ISTART,N
019810                        Y5(J)=Y(I,J)
019820                        EVEXPY(J)=VEXPY(I,J)
019830                        ESEXPY(J)=SEXPY(I,J)
019840    5436              CONTINUE
019850                    DO 5437 J=1,INX1
019860                      DO 5437 JJ=1,INX1
019870                        ECC1(J,JJ)=CC1(J,JJ)
019880    5437              CONTINUE
019890                    EREGSS=0.
019900                    DO 5438 J=1,INX
019910                        EBMAT(J,1)=BBMAT(J+1,1)
019920                        EREGSS=EREGSS+EBMAT(J,1)*FNSP(J,I,X,Y)
019930    5438              CONTINUE
019940                    ERESSS=FNSSY(I,Y)-EREGSS
019950                    EAR=EREGSS/FNSSY(I,Y)
019960                ENDIF
019970                DFAC(I)=DFAC(I) - STEP
019980                IF (DFAC(I).LT.DMIN.OR.DFAC(I).LE.0) GOTO 1001
019990    1000      CONTINUE
020000    1001      CONTINUE
020010 C
020020 C-----FIND BEST SOLUTION, IF THERE WAS ONE, AND OUTPUT THE VALUES.
020030 C
020040            IF (ISTEP.GT.0) THEN
020050                DFAC(I)=EDFAC
020060                SDM=EMIN
020070                VPARMM(I)=EVPARM
020080                A1A=EA1A
020090                ACEPT=EACEPT
020100                RBERT=ERBERT
020110                REGSS=EREGSS
020120                RESSS=ERESSS
020130                AR=EAR
020140                RESMS=RESSS/IRESSS
020150                DO 5439 J=ISTART,N
020160                    Y(I,J)=Y5(J)
```

```
020170                      VEXPY(I,J)=EVEXPY(J)
020180                      SEXPY(I,J)=ESEXPY(J)
020190  5439              CONTINUE
020200                    DO 5440 J=1,INX1
020210                      DO 5440 JJ=1,INX1
020220                        CC1(J,JJ)=ECC1(J,JJ)
020230  5440              CONTINUE
020240                    DO 5441 J=1,INX
020250                      BBMAT(J+1,1)=EBMAT(J,1)
020260                      BMAT(J,1)=EBMAT(J,1)
020270  5441              CONTINUE
020280 C
020290 C-----WE HAVE THE BEST SOLUTION.  CALCULATE THE REST OF THE VALUES
020300 C     NEEDED.
020310 C
020320  5434              RSUMSS=0.
020330                    RBERT2(I)=RBERT
020340                    BMAT1=BMAT(1,1)
020350                    REGMS=REGSS/INX
020360                    SCASS=SCASS+RESSS
020370                    ISCA=ISCA+N-ISTART+1-INX1-LNY(I)
020380                    FREG=REGMS/RESMS
020390                    FRACSS=100.*(AR-(FLOAT(INX-1)/FLOAT(IRESSS))*(1.-AR))
020400                    SACEPT=0.
020410                    DO 763 J2=ISTART,N
020420                      I3=J2-1
020430                      X(1,J2)=SN1(I3)
020440                      X(2,J2)=SN2(I3)
020450  763               CONTINUE
020460                    SACEPT=1./FLOAT(KJ2)
020470                    VPARM(I,1)=SACEPT
020480                    VPARM(I,1)=(VPARM(I,1)+FNAX(1,X,I)*FNAX(1,X,I)*CC1(2,2))
020490                    VPARM(I,1)=VPARM(I,1)/HARN1(I)
020500                    SACEPT=SQRT(VPARM(I,1)*RESMS)
020510                    DO 734 J=1,INX
020520                      VPARM(I,J+1)=CC1(J+1,J+1)/HARN1(I)
020530                      SBMAT(J,1)=SQRT(VPARM(I,J+1)*RESMS)
020540                      CONFL(J,1)=BMAT(J,1)-TT1(I)*SBMAT(J,1)
020550                      CONFU(J,1)=BMAT(J,1)+TT1(I)*SBMAT(J,1)
020560                      SSR(J)=BMAT(J,1)**2/CC1(J+1,J+1)
020570                      SR(J,I)=SSR(J)
020580                      RSUMSS=RSUMSS+SSR(J)
020590  734               CONTINUE
020600                    SR1(I)=REGSS-SSR(1)
020610                    PARM(I,1)=ACEPT
020620                    PARM(I,2)=BMAT(1,1)
020630                    PARM(I,3)=RBERT
020640                    VPARM(I,3)=VPARM(I,3)/BBMAT(2,1)**2
020650                    SRBERT=DABS(SBMAT(2,1)/BBMAT(2,1))
020660                    CONAL=ACEPT-TT1(I)*SACEPT
020670                    CONAU=ACEPT+TT1(I)*SACEPT
020680                    RHOL=RBERT-TT1(I)*SRBERT
020690                    RHOU=RBERT+TT1(I)*SRBERT
020700                    SEINF=SACEPT
020710                    SERHO=SRBERT
020720                    QQ1=PARM(I,1)
020730                    QQ2=PARM(I,2)
020740                    QQ3=PARM(I,3)
020750                    QQ4=DFAC(I)
020760                    SUMRS=SUMRS+REGSS
020770                    DO 735 J=1,INX
020780                      FR(J)=SSR(J)/RESMS
020790  735               CONTINUE
020800                    FSR1=SR1(I)/RESMS
020810                    A1A=A1A*100
020820                    WRITE(8,601)
020830                    WRITE(8,3045)A1A
020840  3045              FORMAT(1X,'CONVERGENCE OCCURRED AT ',F10.7,' PERCENT',
020850       >                 ' INTERVALS')
020860                    WRITE(8,601)
020870                    WRITE(8,650)DFAC(I)
020880  650               FORMAT(1X,'D-FACTOR',T19,F14.6)
```

```
020890                WRITE(8,602)
020900                WRITE(8,603)
020910  603           FORMAT(1X,'CONSTANT',T23,'VALUE',T38,'SE',T53,
020920      >               'CONFIDENCE LIMIT'/T52,'LOWER',T67,'UPPER')
020930                WRITE(8,604)PARM(I,1),SEINF,CONAL,CONAU
020940  604           FORMAT(1X,'Y-INFINITY',T18,F14.6,T31,F14.6,T46,F14.6,T59,
020950      >               F14.6)
020960                WRITE(8,608)(BMAT(J,1),SBMAT(J,1),CONFL(J,1),CONFU(J,1),
020970      >                                                    J=1,INX)
020980  608           FORMAT(1X,'SLOPE-B',T18,F14.6,T31,F14.6,T46,F14.6,T59,
020990      >               F14.6/1X,'AUXILIARY-B1',T19,F14.6,T31,F14.6,T46,
021000      >               F14.6,T59,F14.6)
021010  4533          WRITE(8,656)PARM(I,3),SERHO,RHOL,RHOU
021020  656           FORMAT(1X,'RHO ',T18,F14.6,T31,F14.6,T46,F14.6,T59,F14.6)
021030                WRITE(8,602)
021040 C
021050 C-----GET AND PRINT VALUES FOR 'YINF', 'K', AND 'T0'.
021060 C
021070                CALL MISRA6(QQ1,QQ2,QQ3,QQ4)
021080                WRITE(8,602)
021090                WRITE(8,605)
021100  605           FORMAT(10X,'ANOVA OF MULTIPLE REGRESSION')
021110                WRITE(8,601)
021120                WRITE(8,606)
021130  606           FORMAT(1X,'ITEM',T28,'DF',T39,'MS',T57,'F VALUE')
021140                WRITE(8,601)
021150                WRITE(8,607)INX,REGMS,FREG
021160  607           FORMAT(1X,'ALL SLOPES (B AND B1)',T26,I3,T34,F14.6,T54,
021170      >               F10.3)
021180                WRITE(8,609)(SSR(J),FR(J),J=1,INX)
021190  609           FORMAT(1X,'SLOPE (B)',T28,'1',T34,F14.6,T54,F10.3/1X,
021200      >               'AUXILIARY (B1)',T28,'1',T34,F14.6,T54,F10.3)
021210                WRITE(8,610)IRESSS,RESMS,FRACSS
021220  610           FORMAT(1X,'ERROR',T24,I5,T34,F14.6
021230      >               //1X,'BERT. FUNCTION EXPLAINS ',F5.2,' PER CENT'
021240      >               /1X,'OF THE TOTAL VARIATION AMONG AGES')
021250                WRITE(8,600)
021260                WRITE(8,635)
021270  635           FORMAT(1X,'AT AGE',T10,'GROUP SIZE',T23,'OBSERVED LENGTH'
021280      >               ,T40,'EXPECTED LENGTH',T60,'DIFFERENCE')
021290                WRITE(8,601)
021300                IN4=0
021310                SDM=0.
021320                DO 778 J=ISTART,N
021330                    IF(W1(I,J).NE.0.) THEN
021340                        J1=J-1
021350                        A=SN3(J1)
021360                        DIFF(I,J)=Y(I,J)-A
021370                        IF(VEXPY(I,J).LT.0.) THEN
021380                            WRITE(8,643)J1,N1(I,J),Y(I,J),A
021390  643                       FORMAT(1X,I3,T8,I5,T20,F14.4,T38,F14.4,T56
021400      >                            ,'IS DIFF. SIGNIFICANT?')
021410                          ELSE
021420                            IF(Y(I,J).GT.0.) THEN
021430                                IN4=IN4+1
021440                                BB1=ABS(DIFF(I,J))
021450                                SDM=SDM+BB1/SEXPY(I,J)
021460                                A1=TT1(I)*SEXPY(I,J)
021470                                IF(DIFF(I,J).GT.(-A1).AND.
021480      >                            DIFF(I,J).LT.(A1)) THEN
021490                        WRITE(8,641)J1,N1(I,J),Y(I,J),A
021500  641                   FORMAT(1X,I3,T8,I5,T20,F14.4,T38,F14.4)
021510                              ELSE
021520                        WRITE(8,636)J1,N1(I,J),Y(I,J),A,DIFF(I,J)
021530  636                   FORMAT(1X,I3,T8,I5,T20,F14.4,T38,F14.4,T56,F14.4)
021540                              ENDIF
021550                            ENDIF
021560                        ENDIF
021570                    ENDIF
021580  778           CONTINUE
021590                AIN4=IN4
021600                SDM=SDM/AIN4
```

```
021610               WRITE(8,601)
021620               WRITE(8,5421)SDM
021630  5421         FORMAT(1X,'MEAN STANDARDIZED DIFFERENCE IS ',F14.4)
021640               WRITE(8,601)
021650               DD=1./DFAC(I)
021660               DO 789 J=1,INY
021670                 DO 789 J1=ISTART,N
021680                   W1(J,J1)=N1(J,J1)
021690  789          CONTINUE
021700 C
021710 C-----RE-ESTIMATE 'T0', 'K', AND 'YINF' USING ALLEN'S ALGORITHM.
021720 C
021730               CALL MISRA7(I)
021740               DO 790 J=1,INY
021750                 DO 790 J1=ISTART,N
021760                   W1(J,J1)=W3(J,J1)
021770  790          CONTINUE
021780              ELSE
021790  842          WRITE(8,619)
021800  619          FORMAT(1X,'D-FACTOR IS NOW AT MINIMUM')
021810            ENDIF
021820  701   CONTINUE
021830  812   WRITE(8,601)
021840  2112  DO 300 I=1,INY
021850            DO 300 J=1,N2
021860               Y(I,J)=Y4(I,J)
021870   300  CONTINUE
021880 C
021890 C-----IF THE 'D' VALUES ARE THE SAME, THEN COMPARE THE PARAMETERS.
021900 C
021910        DO 2008 I=2,INY
021920           IF(DFAC(I).NE.DFAC(1))GOTO 2009
021930  2008  CONTINUE
021940        IF(DFAC(1).LE.0.)GOTO 2009
021950        CALL MISRA2
021960  2009  WRITE(8,601)
021970        RETURN
021980        END
021990        SUBROUTINE MISRA2
022000 CC
022010 CC----DESCRIPTION:  THIS SUBROUTINE FITS THE COMBINED SAMPLE USING
022020 CC                  THE SAME METHOD AS USED IN 'MISRA1' EXCEPT THAT
022030 CC                  THE VALUE OF 'D' IS NOT ADJUSTED.
022040 CC
022050 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
022060 CC                         WYCOVE SYSTEMS LIMITED,
022070 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
022080 CC                         RESEARCH BRANCH, DFO, HALIFAX.
022090 CC
022100 CC----DATE:  MARCH 1984.
022110 CC
022120 CC----REVISION HISTORY:  NONE.
022130 CC
022140 CC----FILES:
022150 CC
022160 CC    UNIT    I/O    DESCRIPTION
022170 CC    ------------------------
022180 CC      8      O     RESULTS.
022190 CC
022200 CC----EXTERNALS:
022210 CC
022220 CC    NAME     DESCRIPTION
022230 CC    ------------------
022240 CC    MISRA5   GETS INITIAL ESTIMATE OF 'YINF'.
022250 CC    MISRA6   GETS ESTIMATES OF 'K' AND 'T0'.
022260 CC    TRY1     INITIAL ESTIMATE OF 'RHO' USING WALFORD LINE.
022270 CC    TRY2     INITIAL ESTIMATE OF 'RHO' USING 1ST AND MIDDLE AGE
022280 CC             GROUPS.
022290 CC    TRY3     INITIAL ESTIMATE OF 'RHO' USING LOGARITHMIC FIT
022300 CC             (ONLY 1 OF 'TRY1', 'TRY2' OR 'TRY3' USED).
022310 CC    SMPY     MULTIPLIES SCALAR TIMES MATRIX.
022320 CC    MINV     MATRIX INVERSE.
```

```
022330 CC     GMPRD    MATRIX PRODUCT.
022340 CC     FUNCTIONS  FNX1, FNX2, FNY1, FNXY, FNAX, FNAY, FNSP, FNSSX, FNSSY.
022350 CC
022360 CC----COMMON BLOCKS:
022370 CC
022380 CC     COMMON /AA/
022390 CC
022400 CC     VARIABLE  DESCRIPTION
022410 CC     --------------------
022420 CC     ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
022430 CC               THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
022440 CC     N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
022450 CC               THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
022460 CC     N2        N + 1.
022470 CC     W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
022480 CC     W3(10,30) TEMPORARY COPY OF 'W1(,)'.
022490 CC     LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
022500 CC     LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
022510 CC
022520 CC     COMMON /ONER/
022530 CC
022540 CC     VARIABLE  DESCRIPTION
022550 CC     --------------------
022560 CC     IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
022570 CC               ESTIMATE OF 'RHO'.
022580 CC     TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
022590 CC     TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
022600 CC               DATA.
022610 CC     Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
022620 CC               AGE + 1.
022630 CC     DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
022640 CC     N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
022650 CC     SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
022660 CC     HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
022670 CC     IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
022680 CC               VALUE FOR 'B1/B'.
022690 CC     FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
022700 CC     IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
022710 CC               OF YEAR SAMPLE COLLECTED.
022720 CC     NWT       TYPE OF WEIGHTING:
022730 CC               1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
022740 CC               2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
022750 CC               3 - CONSTANT = 1.
022760 CC     NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
022770 CC               IF 'NTRY'=1 USE 'TRY1', ETC.
022780 CC
022790 CC     COMMON /FNX/
022800 CC
022810 CC     VARIABLE  DESCRIPTION
022820 CC     --------------------
022830 CC     INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
022840 CC     INX1      INX + 1.
022850 CC
022860 CC     COMMON /FNY/
022870 CC
022880 CC     VARIABLE  DESCRIPTION
022890 CC     --------------------
022900 CC     INY       NUMBER OF SAMPLES.
022910 CC     INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
022920 CC
022930 CC     COMMON /AA1/
022940 CC
022950 CC     VARIABLE  DESCRIPTION
022960 CC     --------------------
022970 CC     RBERT1(10)   INITIAL ESTIMATE OF 'RHO' FROM 'TRY1' (2 OR 3).
022980 CC
022990 CC     COMMON /AA2/
023000 CC
023010 CC     VARIABLE  DESCRIPTION
023020 CC     --------------------
023030 CC     ITRY      CONTROLS HEADINGS IN 'TRY' ROUTINES.
023040 CC     AYINF     IN 'TRY2' AND 'TRY3', SET EQUAL TO NUMBER OF TIMES
```

```
023050 CC              ESTIMATE OF 'YINF' HAS TO BE MULTIPLIED BY 'FACK()'
023060 CC              TO GET A REASONABLE VALUE OF 'RHO'.
023070 CC
023080 CC     COMMON /AA3/
023090 CC
023100 CC     VARIABLE  DESCRIPTION
023110 CC     --------------------
023120 CC     YINF1      INITIAL ESTIMATE OF 'YINF' SET IN 'MISRA5'.
023130 CC
023140 CC     COMMON /AA4/
023150 CC
023160 CC     VARIABLE  DESCRIPTION
023170 CC     --------------------
023180 CC     RBERT2(10)   FINAL ESTIMATE OF 'RHO' FROM 'MISRA1'.
023190 CC
023200 CC     COMMON /AA10/
023210 CC
023220 CC     VARIABLE  DESCRIPTION
023230 CC     --------------------
023240 CC     FACK(10)  FACTOR FOR INCREASING ESTIMATE OF 'YINF' IN 'TRY2' OR
023250 CC               'TRY3'.
023260 CC     KN1       MAXIMUM NUMBER OF ITERATIONS OF LOOP THAT RE-ESTIMATES
023270 CC               'YINF' BY 'TRY' ROUTINES, THEN REDOES REGRESSION.
023280 CC
023290 CC     COMMON /AA11/
023300 CC
023310 CC     VARIABLE  DESCRIPTION
023320 CC     --------------------
023330 CC     DFAC(10)  'D' VALUE FOR SAMPLE.
023340 CC     DMAX      STARTING (MAXIMUM) VALUE FOR 'D'.
023350 CC     STEP      VALUE SUBTRACTED FROM 'D' EACH ITERATION.
023360 CC     DMIN      MINIMUM VALUE FOR 'D'.
023370 CC     Y4(10,30) TEMPORARY STORAGE FOR 'Y(,)'.
023380 CC
023390 CC     COMMON /AA15/
023400 CC
023410 CC     VARIABLE  DESCRIPTION
023420 CC     --------------------
023430 CC     FCEPT     ACCEPTANCE CONTROL ON 'YINF', MUST BE LESS THAN 'FCEPT'
023440 CC               TIMES REVISED ESTIMATE OF 'YINF' FROM 'TRY' ROUTINES.
023450 CC
023460 CC     COMMON /AA19/
023470 CC
023480 CC     VARIABLE  DESCRIPTION
023490 CC     --------------------
023500 CC     DETC1     MIN VALUE ACCEPTABLE FOR DETERMINANT IF INVERTING
023510 CC               MATRIX.
023520 CC     DETC2     MAX VALUE ACCEPTABLE FOR DETERMINANT IF INVERTING
023530 CC               MATRIX.
023540 CC     ZFAC      FACTOR FOR SCALING MATRIX IF DETERMINANT TOO BIG OR
023550 CC               SMALL.
023560 CC
023570 CC     COMMON /COM/
023580 CC
023590 CC     VARIABLE  DESCRIPTION
023600 CC     --------------------
023610 CC     SCASSE    VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
023620 CC               ANOVA.
023630 CC     ISCAE     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
023640 CC               ANOVA.
023650 CC     SCASS     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
023660 CC               ANOVA.
023670 CC     ISCA      VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
023680 CC               ANOVA.
023690 CC     SUMRSS    VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
023700 CC               ANOVA.
023710 CC     COMSS     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
023720 CC               ANOVA.
023730 CC     SR(2,10)  VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
023740 CC               ANOVA.
023750 CC     SR1(10)   VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
023760 CC               ANOVA.
```

```
023770 CC
023780 CC     COMMON /CCM/
023790 CC
023800 CC     VARIABLE  DESCRIPTION
023810 CC     --------------------
023820 CC     PARM(10,3)   FIT PARAMETERS FOR EACH SAMPLE:  'YINF', 'B', 'RHO'.
023830 CC     VPARM(10,3)  VARIANCE ESTIMATES FOR FIT PARAMETERS.
023840 CC
023850 CC----STATEMENT FUNCTIONS:
023860 CC
023870 CC     PROTOTYPE  DESCRIPTION
023880 CC     --------------------
023890 CC     SN1(T)     'RHO**T' (THIS IS 'Z' IN MISRA PAPER).
023900 CC     SN2(T)     'T*RHO**(T-1)' (THIS IS 'Z1' IN MISRA PAPER).
023910 CC     SN3(T)     PREDICTED LENGTH.
023920 CC
023930       DOUBLE PRECISION ZFAC1,ZFAC2,CRIT,DCC1,BBMAT(3,1),
023940      >                 CC1(3,3),XY1(3,1)
023950       SN1(K)=RBERT**K
023960       SN2(K)=K*SN1(K)/RBERT
023970       SN3(K)=ACEPT+BMAT1*SN1(K)
023980 C
023990       COMMON /AA/    ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
024000       COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
024010      >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NW1,NTRY
024020       COMMON /FNX/   INX,INX1
024030       COMMON /FNY/   INY,INY1
024040       COMMON /AA1/   RBERT1(10)
024050       COMMON /AA2/   ITRY,AYINF
024060       COMMON /AA3/   YINF1
024070       COMMON /AA4/   RBERT2(10)
024080       COMMON /AA10/ FACK(10),KN1
024090       COMMON /AA11/ DFAC(10),DMAX,STEP,DMIN,Y4(10,30)
024100       COMMON /AA15/ FCEPT
024110       COMMON /AA19/ DETC1,DETC2,ZFAC
024120       COMMON /COM/   SCASSE,ISCAE,SCASS,ISCA,SUMRSS,COMSS,SR(2,10),
024130      >               SR1(10)
024140       COMMON /CMM/   PARM(10,3),VPARM(10,3)
024150 C
024160 C
024170       DIMENSION BMAT(2,1),SSR(2),FR(2),SBMAT(2,1),CONFL(2,1),CONFU(2,1)
024180       DIMENSION X(3,30),LW(3),MW(3),VEXPY(30),SEXPY(30)
024190 C
024200  601  FORMAT(/)
024210  602  FORMAT(//)
024220 C
024230 C
024240 C-----INITIALIZE VARIABLES.  SET PARAMETERS.
024250       DO 746 I=1,N2
024260           DO 746 J=1,INX
024270               X(J,I)=0.
024280  746  CONTINUE
024290       DFAC(INY1)=DFAC(1)
024300       ZFAC2=ZFAC
024310       SW=0.
024320       DO 701 I=ISTART,N
024330           SW=SW+W1(INY1,I)
024340  701  CONTINUE
024350       INN2=IN2(INY1)
024360       INN3=IN3(INY1)
024370       AYINF=1.
024380 C
024390 C
024400 C-----THIS TIME DO NOT LOOP ON VALUES FOR 'D', JUST USE VALUE OF
024410 C     'D' FOUND FOR INDIVIDUAL SAMPLES.  FIRST PUT AVERAGE OF THE
024420 C     TRANSFORMED AVERAGES IN 'Y(INY1,.)'.  THEN CALL 'MISRA5' AS
024430 C     BEFORE.
024440       DO 799 I=1,INY
024450         DO 799 J=ISTART,N
024460           Y(I,J)=Y4(I,J)
024470           IF (Y(I,J).GT.0.0001) THEN
024480               Y(I,J)=Y(I,J)**DFAC(INY1)
```

```
024490           ENDIF
024500  799  CONTINUE
024510       DO 3077 I=ISTART,N
024520           Y(INY1,I)=0
024530           DO 3078 J=1,INY
024540               Y(INY1,I)=Y(INY1,I)+Y(J,I)
024550  3078     CONTINUE
024560           KA=INY-LNA(I)
024570           A=KA
024580           IF (KA.NE.0) Y(INY1,I)=Y(INY1,I)/A
024590  3077 CONTINUE
024600       CALL MISRA5(INY1)
024610 C
024620 C
024630 C-----LOOP ON INCREMENTED 'YINF' VALUE.  RE-ESTIMATE 'RHO' AND TRY
024640 C     AGAIN.  ONLY LOOP 'KN1' TIMES.  'KN1' CAN BE SET BY USER FOR
024650 C     'TRY2' AND 'TRY3'.  FOR 'TRY1' IT IS HELD TO 1.
024660       DO 990 IT=1,KN1
024670           IF (NTRY.EQ.1) THEN
024680               CALL TRY1(INY1)
024690             ELSE IF (NTRY.EQ.2) THEN
024700               CALL TRY2(INY1)
024710             ELSE
024720               CALL TRY3(INY1)
024730           ENDIF
024740           RBERT=RBERT1(INY1)
024750           ACEPT1=YINF1*AYINF
024760           ACEPT1=FCEPT*ACEPT1
024770 C
024780 C
024790 C-----LOOP TO CONTROL THE PRECISION OF THE TEST ON 'B1/B'.
024800 C     TEST VALUE IS SET TO 'FACK4()' AT FIRST AND IS
024810 C     INCREMENTED BY 'FACK4()' EACH ITERATION.  MAX NUMBER
024820 C     OF ITERATIONS COMES FROM 'IN3()'.  'FACK4()' AND
024830 C     'IN3()' CAN BE SET BY USER.
024840           A1A=0.
024850           DO 980 I1=1,INN3
024860               A1A=A1A+FACK4(INY1)
024870 C
024880 C
024890 C-----LOOP TO IMPROVE ESTIMATE OF 'RHO' BY ADDING CALCULATED
024900 C     VALUE OF 'B1/B'.  CUT OFF IF LESS THAN VALUE SET IN
024910 C     980 LOOP.  MAX NUMBER OF ITERATIONS COMES FROM 'IN2()'
024920 C     WHICH CAN BE SET BY USER.
024930                   DO 970 IIN2=1,INN2
024940 C
024950 C
024960 C-----'X(,)' CONTAINS 'Z' AND 'Z1'.
024970                       DO 753 I2=ISTART,N
024980                           I3=I2-1
024990                           X(1,I2)=SN1(I3)
025000                           X(2,I2)=SN2(I3)
025010  753                  CONTINUE
025020                       DO 4156 J=1,INX
025030                           XY1(J+1,1)=FNSP(J,INY1,X,Y)
025040  4156                 CONTINUE
025050                       DO 4157 II1=2,INX1
025060                         DO 4157 J=2,INX1
025070                             CC1(II1,J)=FNSSX(II1-1,J-1,X,INY1)
025080  4157                 CONTINUE
025090                       A1BB=CC1(3,3)*XY1(2,1)-CC1(2,3)*XY1(3,1)
025100                       A1B=CC1(2,2)*XY1(3,1)-CC1(2,3)*XY1(2,1)
025110 C
025120 C
025130 C-----EQUATION 16.35, P.149 OF BLISS.  THIS IS 'B1/B'.
025140                       A1B=A1B/A1BB
025150                       A1BB=ABS(A1B)
025160                       IF (A1BB.LE.A1A) THEN
025170 C
025180 C
025190 C-----IF 'B1/B' IS SMALL, THEN WORK OUT THE PARAMETERS OF THE EQUATION
025200 C     AND CHECK IF FIT IS OKAY.  SEE BLISS P.147-149.
```

```
025210                    DO 702 J=1,INX
025220                        XY1(J+1,1)=FNXY(J,INY1,X,Y)
025230  702               CONTINUE
025240 C
025250 C
025260 C-----THIS LOOP HANDLES INVERSE OF COEF. MATRIX OF NORMAL EQUATIONS.
025270 C     IF DETERMINANT IS TOO EXTREME, THEN MATRIX IS SCALED.
025280 C     'SMPY' IS SCALAR MULT. 1ST TERM * 2ND, PUT IN 3RD.
025290 C     'GMPRD' IS MATRIX PROD. 1ST TERM * 2ND, PUT IN 3RD.
025300 C     'MINV' IS INVERSE OF 1ST TERM, RETURN IN 1ST, DETERMINANT IN 2ND.
025310                    DO 4036 IINV=1,100
025320                        CC1(1,1)=SW
025330                        DO 719 I=2,INX1
025340                            CC1(1,I)=FNX1(I-1,X,INY1)
025350                            CC1(I,1)=CC1(1,I)
025360                          DO 719 J=2,INX1
025370                            CC1(I,J)=FNX2(I-1,J-1,X,INY1)
025380  719               CONTINUE
025390                    XY1(1,1)=FNY1(INY1,Y)
025400                    CALL SMPY(CC1,ZFAC2,CC1,INX1,INX1,0)
025410                    DCC1=0.
025420                    CALL MINV(CC1,INX1,DCC1,LW,MW)
025430                    CALL SMPY(CC1,ZFAC2,CC1,INX1,INX1,0)
025440                    IF (DCC1.LE.DETC1) THEN
025450                        ZFAC2=ZFAC2*2.14
025460                      ELSE IF (DCC1.GE.DETC2) THEN
025470                        ZFAC2=ZFAC2*0.47
025480                      ELSE
025490                        GOTO 4035
025500                    ENDIF
025510  4036              CONTINUE
025520  4035              CALL GMPRD(CC1,XY1,BBMAT,INX1,INX1,1)
025530 C
025540 C
025550 C-----'ACEPT2' IS CALCULATED VALUE OF 'YINF'.  IF THIS VALUE IS LESS
025560 C     THAN ZERO OR IS GREATER THAN 'ACEPT1', THIS SOLUTION IS REJECTED.
025570 C     'ACEPT1' IS 'YINF' AFTER 'TRY1' (2 OR 3) * 'FCEPT', A FACTOR
025580 C     THAT CAN BE SET BY THE USER.  SOLUTION IS ALSO REJECTED IF THE
025590 C     ESTIMATE OF 'BETA' ('BBMAT(2,1)') IS POSITIVE, OR IF 'B1'
025600 C     ('BBMAT(3,1)') IS MORE THAN 'TT1()' * (1 STANDARD ERROR OF 'B1')
025610 C     FROM ZERO.  'TT1()' CAN BE SET BY THE USER.
025620                    ACEPT2=FNAY(INY1,Y)-BBMAT(2,1)*FNAX(1,X,INY1)
025630                    IF (ACEPT2.LE.0.) GOTO 969
025640                    IF (ACEPT2.GT.ACEPT1) GOTO 969
025650                    IF (BBMAT(2,1).GT.0.) GOTO 969
025660                    RSSSA=FNSSY(INY1,Y)-BBMAT(2,1)*FNSP(1,INY1,X,Y)
025670       >                  -BBMAT(3,1)*FNSP(2,INY1,X,Y)
025680                    RSMSA=N-ISTART-INX-LNY(INY1)
025690                    RSMSA=RSSSA/RSMSA
025700                    CRIT=DSQRT((CC1(3,3)/HARN1(INY1))*RSMSA)
025710                    CRIT=TT1(INY1)*CRIT
025720                    IF (DABS(BBMAT(3,1)).GT.DABS(CRIT)) GOTO 969
025730                    GOTO 999
025740                  ELSE
025750  969               RBERT=RBERT+A1B
025760                    IF (RBERT.GE.1.) GOTO 990
025770                ENDIF
025780  970             CONTINUE
025790  980         CONTINUE
025800  990   CONTINUE
025810 C
025820 C
025830 C-----IF WE FINISH THE 990 LOOP WITHOUT JUMPING TO 999 THEN THERE IS
025840 C     NO ACCEPTABLE SOLUTION FOR THIS VALUE OF 'DFAC()', SO GOTO 1000.
025850        GOTO 1000
025860 C
025870 C
025880 C-----JUMPS TO 999 IF SOLUTION IS ACCEPTABLE.  NOW ESTIMATE PARAMETERS.
025890 C     'ACEPT' IS ESTIMATE OF 'YINF'.  'REGSS' IS SUM OF SQUARES
025900 C     ATTRIBUTABLE TO REGRESSION, 16.44 BLISS.  'RESMS' IS ERROR
025910 C     VARIANCE, 16.45 BLISS.  'BMAT1' IS ESTIMATE OF 'RHO'.  'VEXPY(,)'
025920 C     IS VARIANCE OF THE PREDICTED RESPONSE, 16.50 BLISS.
```

```
025930  999  REGSS=0.
025940       RBERT2(INY1)=RBERT
025950       RSUMSS=0.
025960       ACEPT=FNAY(INY1,Y)-BBMAT(2,1)*FNAX(1,X,INY1)
025970       DO 703 J=1,INX
025980           BMAT(J,1)=BBMAT(J+1,1)
025990           REGSS=REGSS+BMAT(J,1)*FNSP(J,INY1,X,Y)
026000  703  CONTINUE
026010       BMAT1=BMAT(1,1)
026020       REGMS=REGSS/INX
026030       RESSS=FNSSY(INY1,Y)-REGSS
026040       SCASSE=SCASS
026050       ISCAE=ISCA
026060       SCASS=SCASS-RESSS
026070       ISCA=ISCA-(N-ISTART+1-INX1-LNY(INY1))
026080       IRESSS=N-ISTART+1-INX-1-LNY(INY1)
026090       RESMS=RESSS/IRESSS
026100       FREG=REGMS/RESMS
026110       AR=REGSS/FNSSY(INY1,Y)
026120       FRACSS=100.*(AR-(FLOAT(INX-1)/FLOAT(IRESSS))*(1.-AR))
026130       SACEPT=0.
026140       KJ2=0
026150       DO 769 J2=ISTART,N
026160           IF(W1(INY1,J2).EQ.0.) THEN
026170               KJ2=KJ2+1
026180           ENDIF
026190  769  CONTINUE
026200       KJ2=N-ISTART+1-KJ2
026210       SACEPT=1./FLOAT(KJ2)
026220       VPARMM=SACEPT
026230       VPARM(INY1,1)=SACEPT
026240       COMSS=REGSS
026250       SUMRSS=SUMRS-COMSS
026260       VPARM(INY1,1)=VPARM(INY1,1)+FNAX(1,X,INY1)
026270      >              *FNAX(1,X,INY1)*CC1(2,2)
026280       DO 3048 J=1,INX
026290         DO 3048 J1=1,INX
026300           VPARMM=VPARMM+FNAX(J,X,INY1)*FNAX(J1,X,INY1)*CC1(J+1,J1+1)
026310 3048  CONTINUE
026320       VPARMM=VPARMM/HARN1(INY1)
026330       VPARM(INY1,1)=VPARM(INY1,1)/HARN1(INY1)
026340       SACEPT=SQRT(VPARM(INY1,1)*RESMS)
026350       DO 711 J=1,INX
026360           VPARM(INY1,J+1)=CC1(J+1,J+1)/HARN1(INY1)
026370           SBMAT(J,1)=SQRT(VPARM(INY1,J+1)*RESMS)
026380           CONFL(J,1)=BMAT(J,1)-TT1(INY1)*SBMAT(J,1)
026390           CONFU(J,1)=BMAT(J,1)+TT1(INY1)*SBMAT(J,1)
026400           SSR(J)=BMAT(J,1)**2/CC1(J+1,J+1)
026410           SR(J,INY1)=SSR(J)
026420           RSUMSS=RSUMSS+SSR(J)
026430  711  CONTINUE
026440       SR1(INY1)=REGSS-SSR(1)
026450       DO 784 J=ISTART,N
026460           VEXPY(J)=X(1,J)*(2.*CC1(1,2)+X(1,J)*CC1(2,2)+2.*X(2,J)
026470      >             *CC1(2,3))
026480           VEXPY(J)=VEXPY(J)+X(2,J)*(2.*CC1(1,3)+X(2,J)*CC1(3,3))
026490           VEXPY(J)=(VPARMM+VEXPY(J)/HARN1(INY1))*RESMS
026500           IF(VEXPY(J).GE.0.) THEN
026510               SEXPY(J)=SQRT(VEXPY(J))
026520           ENDIF
026530  784  CONTINUE
026540       PARM(INY1,1)=ACEPT
026550       PARM(INY1,2)=BMAT(1,1)
026560       PARM(INY1,3)=RBERT
026570       VPARM(INY1,3)=VPARM(INY1,3)/BBMAT(2,1)**2
026580       SRBERT=DABS(SBMAT(2,1)/BBMAT(2,1))
026590       CONAL=ACEPT-TT1(INY1)*SACEPT
026600       CONAU=ACEPT+TT1(INY1)*SACEPT
026610       RHOL=RBERT-TT1(INY1)*SRBERT
026620       RHOU=RBERT+TT1(INY1)*SRBERT
026630       SEINF=SACEPT
026640       SERHO=SRBERT
```

```
026650       QQ1=PARM(INY1,1)
026660       QQ2=PARM(INY1,2)
026670       QQ3=PARM(INY1,3)
026680       QQ4=DFAC(INY1)
026690       DO 735 J=1,INX
026700           FR(J)=SSR(J)/RESMS
026710  735  CONTINUE
026720       FSR1=SR1(INY1)/RESMS
026730 C
026740 C
026750 C-----GET AND PRINT VALUES FOR 'YINF', 'K', AND 'T0'.
026760       WRITE(8,'(1H1,//A,//)') '  FOR THE COMBINED SAMPLE'
026770       CALL MISRA6(QQ1,QQ2,QQ3,QQ4)
026780       WRITE(8,601)
026790       RETURN
026800 C
026810 C
026820  1000 CONTINUE
026830 C-----COMES HERE IF DID NOT FIND A FIT.
026840   842  WRITE(8,619) INN3
026850   619  FORMAT(1X,'WATCH IT--NUMBER OF ITERATIONS IS ',I4,' NOW')
026860       WRITE(8,601)
026870       RETURN
026880       END
026890 C
026900 C*************************************************************************
026910 C
026920       SUBROUTINE MISRA3
026930 CC
026940 CC----DESCRIPTION:  COMPARE THE SAMPLES BY ANOVA AND CHI-SQUARES.
026950 CC
026960 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
026970 CC                          WYCOVE SYSTEMS LIMITED,
026980 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
026990 CC                          RESEARCH BRANCH, DFO, HALIFAX.
027000 CC
027010 CC----DATE:  MARCH 1984.
027020 CC
027030 CC----REVISION HISTORY:  NONE.
027040 CC
027050 CC----FILES:
027060 CC
027070 CC    UNIT    I/O    DESCRIPTION
027080 CC    ------------------------
027090 CC      8      O     RESULTS.
027100 CC
027110 CC----COMMON BLOCKS:
027120 CC
027130 CC    COMMON /AA/
027140 CC
027150 CC    VARIABLE  DESCRIPTION
027160 CC    --------------------
027170 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
027180 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
027190 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
027200 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
027210 CC    N2        N + 1.
027220 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
027230 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
027240 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
027250 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
027260 CC
027270 CC    COMMON /ONER/
027280 CC
027290 CC    VARIABLE  DESCRIPTION
027300 CC    --------------------
027310 CC    IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
027320 CC              ESTIMATE OF 'RHO'.
027330 CC    TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
027340 CC    TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
027350 CC              DATA.
027360 CC    Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
```

```
027370 CC              AGE + 1.
027380 CC     DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
027390 CC     N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
027400 CC     SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
027410 CC     HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
027420 CC     IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
027430 CC               VALUE FOR 'B1/B'.
027440 CC     FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
027450 CC     IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
027460 CC               OF YEAR SAMPLE COLLECTED.
027470 CC     NWT       TYPE OF WEIGHTING:
027480 CC               1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
027490 CC               2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
027500 CC               3 - CONSTANT = 1.
027510 CC     NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
027520 CC               IF 'NTRY'=1 USE 'TRY1', ETC.
027530 CC
027540 CC     COMMON /FNX/
027550 CC
027560 CC     VARIABLE  DESCRIPTION
027570 CC     --------------------
027580 CC     INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
027590 CC     INX1      INX + 1.
027600 CC
027610 CC     COMMON /FNY/
027620 CC
027630 CC     VARIABLE  DESCRIPTION
027640 CC     --------------------
027650 CC     INY       NUMBER OF SAMPLES.
027660 CC     INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
027670 CC
027680 CC     COMMON /AA11/
027690 CC
027700 CC     VARIABLE  DESCRIPTION
027710 CC     --------------------
027720 CC     DFAC(10)  'D' VALUE FOR SAMPLE.
027730 CC     DMAX      STARTING (MAXIMUM) VALUE FOR 'D'.
027740 CC     STEP      VALUE SUBTRACTED FROM 'D' EACH ITERATION.
027750 CC     DMIN      MINIMUM VALUE FOR 'D'.
027760 CC     Y4(10,30) TEMPORARY STORAGE FOR 'Y(,)'.
027770 CC
027780 CC     COMMON /COM/
027790 CC
027800 CC     VARIABLE  DESCRIPTION
027810 CC     --------------------
027820 CC     SCASSE    VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
027830 CC               ANOVA.
027840 CC     ISCAE     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
027850 CC               ANOVA.
027860 CC     SCASS     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
027870 CC               ANOVA.
027880 CC     ISCA      VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
027890 CC               ANOVA.
027900 CC     SUMRSS    VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
027910 CC               ANOVA.
027920 CC     COMSS     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
027930 CC               ANOVA.
027940 CC     SR(2,10)  VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
027950 CC               ANOVA.
027960 CC     SR1(10)   VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
027970 CC               ANOVA.
027980 CC
027990 CC
028000       COMMON /AA/    ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
028010       COMMON /ONER/  IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
028020      >               SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
028030       COMMON /FNX/   INX,INX1
028040       COMMON /FNY/   INY,INY1
028050       COMMON /AA11/  DFAC(10),DMAX,STEP,DMIN,Y4(10,30)
028060       COMMON /COM/   SCASSE,ISCAE,SCASS,ISCA,SUMRSS,COMSS,SR(2,10),
028070      >               SR1(10)
028080 C
```

```
028090       DIMENSION AGE(30),BCOMSS(2),AYBSS(2),AYBMS(2),FAYB(2),FBCOM(2)
028100       DIMENSION CMAT(2,2)
028110 C
028120  601  FORMAT(/)
028130  602  FORMAT(//)
028140       ISUMR=INX*(INY-1)
028150       SUMRMS=SUMRSS/ISUMR
028160       SWA=0.
028170       SWA1=0.
028180       SWA2=0.
028190       SWY=0.
028200       SWY1=0.
028210       SWY2=0.
028220       AYSR1=0.
028230       DO 2009 I=1,INY
028240         DO 2009 J=ISTART,N
028250           Y(I,J)=Y4(I,J)
028260           IF (Y(I,J).GT.0.0001) Y(I,J)=Y(I,J)**DFAC(I)
028270  2009 CONTINUE
028280       DO 3077 I=1,INY
028290           Y(I,N2)=0.
028300         DO 3077 J=ISTART,N
028310             Y(I,N2)=Y(I,N2)+Y(I,J)
028320  3077 CONTINUE
028330       DO 706 I=1,INY
028340           KA=N-ISTART+1-LNY(I)
028350           A=KA
028360           IF (KA.NE.0) Y(I,N2)=Y(I,N2)/A
028370           SWY=SWY+W1(I,N2)
028380           SWY1=SWY1+Y(I,N2)*W1(I,N2)
028390           SWY2=SWY2+Y(I,N2)**2*W1(I,N2)
028400           AYSR1=AYSR1+SR1(I)
028410  706  CONTINUE
028420       YRSS=SWY2-SWY1**2/SWY
028430       IYR=INY-1
028440       YRMS=YRSS/IYR
028450       DO 707 J=ISTART,N
028460           SWA=SWA+W1(INY1,J)
028470           SWA1=SWA1+W1(INY1,J)*Y(INY1,J)
028480           SWA2=SWA2+W1(INY1,J)*Y(INY1,J)**2
028490  707  CONTINUE
028500       AGESS=SWA2-SWA1**2/SWA
028510       IAGE=N-ISTART-LNY(INY1)
028520       AGEMS=AGESS/IAGE
028530       COMMS=COMSS/INX
028540       SCABSS=AGESS-COMSS
028550       ISCAB=IAGE-INX
028560       SCABMS=SCABSS/ISCAB
028570       SCAMS=SCASS/ISCA
028580       SCAMSE=SCASSE/ISCAE
028590       FSR1=SR1(INY1)/SCAMS
028600       DO 703 I=1,INX
028610           AYBSS(I)=0.
028620           BCOMSS(I)=SR(I,INY1)
028630           FBCOM(I)=BCOMSS(I)/SCAMS
028640           DO 704 J=1,INY
028650               AYBSS(I)=AYBSS(I)+SR(I,J)
028660  704      CONTINUE
028670           AYBSS(I)=AYBSS(I)-BCOMSS(I)
028680           AYBMS(I)=AYBSS(I)/IYR
028690           FAYB(I)=AYBMS(I)/SCAMS
028700  703  CONTINUE
028710       AYSR1=AYSR1-SR1(INY1)
028720       FAYSR1=AYSR1/(SCAMS*FLOAT(IYR))
028730       FSUMR=SUMRMS/SCAMSE
028740       FCOM=COMMS/SCAMS
028750       WRITE(8,603)
028760  603  FORMAT(1H1,///1X,'COMPARISION OF SAMPLES BY ANOVA AND CHI-SQUARE')
028770       WRITE(8,601)
028780       WRITE(8,604)
028790  604  FORMAT(1X,'ITEM',T37,'DF',T50,'MS',T67,'F')
028800       WRITE(8,601)
```

```
028810       WRITE(8,605)INX,COMMS,FCOM
028820  605  FORMAT(1X,'COMMON BERTALANFFY',T34,I5,T44,F14.6,T62,F10.4)
028830       WRITE(8,606)(BCOMSS(I),FBCOM(I),I=1,INX)
028840  606  FORMAT(1X,'COMMON SLOPE ',T38,'1',T44,F14.6,T62,F10.4/1X,
028850      >       'COMMON AUXILIARY ',T38,'1',T44,F14.6,T62,F10.4)
028860       WRITE(8,607)ISUMR,SUMRMS,FSUMR
028870  607  FORMAT(1X,'BERT. CURVES X SAMPLES ',T34,I5,T44,F14.6,T62,F10.4)
028880       WRITE(8,608)(IYR,AYBMS(I),FAYB(I),I=1,INX)
028890  608  FORMAT(1X,'SLOPE X SAMPLES ',T34,I5,T44,F14.6,T62,F10.4/1X,
028900      >         'AUXILIARY X SAMPLES ',T34,I5,T44,F14.6,T62,F10.4)
028910       WRITE(8,4162)ISCA,SCAMS
028920  4162 FORMAT(1X,'SCATTER X SAMPLES',T34,I5,T44,F14.6)
028930       WRITE(8,609)ISCAE,SCAMSE
028940  609  FORMAT(1X,'ERROR',T34,I5,T44,F14.6)
028950       WRITE(8,602)
028960       RETURN
028970       END
028980 C
028990 C*************************************************************************
029000 C
029010       SUBROUTINE TRY1(I)
029020 CC
029030 CC----DESCRIPTION:  DERIVES AN INITIAL VALUE FOR 'RHO' USING THE
029040 CC                  WALFORD LINE.
029050 CC
029060 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
029070 CC                        WYCOVE SYSTEMS LIMITED,
029080 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
029090 CC                        RESEARCH BRANCH, DFO, HALIFAX.
029100 CC
029110 CC----DATE:  MARCH 1984.
029120 CC
029130 CC----REVISION HISTORY:  NONE.
029140 CC
029150 CC----FILES:
029160 CC
029170 CC    UNIT    I/O    DESCRIPTION
029180 CC    ------------------------
029190 CC      8      O     RESULTS.
029200 CC
029210 CC----FORMAL PARAMETERS:
029220 CC
029230 CC    PARAMETER  I/O  DESCRIPTION
029240 CC    -------------------------
029250 CC    I           I   SAMPLE NUMBER.
029260 CC
029270 CC----COMMON BLOCKS:
029280 CC
029290 CC    COMMON /AA/
029300 CC
029310 CC    VARIABLE  DESCRIPTION
029320 CC    --------------------
029330 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
029340 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
029350 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
029360 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
029370 CC    N2        N + 1.
029380 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
029390 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
029400 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
029410 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
029420 CC
029430 CC    COMMON /ONER/
029440 CC
029450 CC    VARIABLE  DESCRIPTION
029460 CC    --------------------
029470 CC    IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
029480 CC              ESTIMATE OF 'RHO'.
029490 CC    TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
029500 CC    TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
029510 CC              DATA.
029520 CC    Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
```

```
029530 CC              AGE + 1.
029540 CC     DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
029550 CC     N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
029560 CC     SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
029570 CC     HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
029580 CC     IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
029590 CC               VALUE FOR 'B1/B'.
029600 CC     FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
029610 CC     IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
029620 CC               OF YEAR SAMPLE COLLECTED.
029630 CC     NWT       TYPE OF WEIGHTING:
029640 CC               1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
029650 CC               2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
029660 CC               3 - CONSTANT = 1.
029670 CC     NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
029680 CC               IF 'NTRY'=1 USE 'TRY1', ETC.
029690 CC
029700 CC     COMMON /AA1/
029710 CC
029720 CC     VARIABLE  DESCRIPTION
029730 CC     --------------------
029740 CC     RBERT1(10)    INITIAL ESTIMATE OF 'RHO' FROM 'TRY1' (2 OR 3).
029750 CC
029760 CC     COMMON /AA2/
029770 CC
029780 CC     VARIABLE  DESCRIPTION
029790 CC     --------------------
029800 CC     ITRY      CONTROLS HEADINGS IN 'TRY' ROUTINES.
029810 CC     AYINF     IN 'TRY2' AND 'TRY3', SET EQUAL TO NUMBER OF TIMES
029820 CC               ESTIMATE OF 'YINF' HAS TO BE MULTIPLIED BY 'FACK()'
029830 CC               TO GET A REASONABLE VALUE OF 'RHO'.
029840 CC
029850 CC     COMMON /AA10/
029860 CC
029870 CC     VARIABLE  DESCRIPTION
029880 CC     --------------------
029890 CC     FACK(10)  FACTOR FOR INCREASING ESTIMATE OF 'YINF' IN 'TRY2' OR
029900 CC               'TRY3'.
029910 CC     KN1       MAXIMUM NUMBER OF ITERATIONS OF LOOP THAT RE-ESTIMATES
029920 CC               'YINF' BY 'TRY' ROUTINES, THEN REDOES REGRESSION.
029930 CC
029940       COMMON /AA/   ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
029950       COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
029960      >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
029970       COMMON /AA1/  RBERT1(10)
029980       COMMON /AA2/  ITRY,AYINF
029990       COMMON /AA10/ FACK(10),KN1
030000 C
030010       DIMENSION Y2(30),W2(30),EXPY(30),DIF(30)
030020 C
030030       ITRY=ITRY+1
030040       IF (ITRY.EQ.1) WRITE(8,5424)
030050  5424 FORMAT(1X,'USING TRY1')
030060 C
030070 C
030080 C-----REMOVE AGES WITH NO DATA.
030090       N3=0
030100       DO 810 IN1=ISTART,N
030110           IF (W1(I,IN1).NE.0.) THEN
030120               N3=N3+1
030130               Y2(N3)=Y(I,IN1)
030140               W2(N3)=W1(I,IN1)
030150           ENDIF
030160   810 CONTINUE
030170       SW=0.
030180       SWX=0.
030190       SWY=0.
030200       SWXY=0.
030210       SWXX=0.
030220       N4=N3-1
030230 C
030240 C
```

```
030250 C-----FIT EQUATION  'Y(T+1)=A+RHO*Y(T)'.
030260       DO 700 K=1,N4
030270          SWX=SWX+W2(K+1)*Y2(K)
030280          SW=SW+W2(K+1)
030290          SWXX=SWXX+W2(K+1)*Y2(K)**2
030300          SWY=SWY+W2(K+1)*Y2(K+1)
030310          SWXY=SWXY+W2(K+1)*Y2(K)*Y2(K+1)
030320   700 CONTINUE
030330       AR1=SWXY-SWX*SWY/SW
030340       AR2=SWXX-SWX*SWX/SW
030350       RBERT1(I)=AR1/AR2
030360       WRITE(8,603)I,RBERT1(I)
030370   603 FORMAT(1X,'WALFORD(',I2,') IS ',F10.6)
030380       YBAR=SWY/SW
030390       XBAR=SWX/SW
030400       ACEPT=YBAR-RBERT1(I)*XBAR
030410       WRITE(8,601)
030420   601 FORMAT(/)
030430       WRITE(8,605)
030440   605 FORMAT(1X,'AT AGE',T14,'OBSERVED LENGTH',T32,'EXPECTED LENGTH',
030450      >         T55,'DIFFERENCE')
030460       WRITE(8,601)
030470       DO 3040 J=ISTART,N
030480          DIF(J)=0.
030490  3040 CONTINUE
030500       NN4=N-1
030510       DO 702 J=ISTART,NN4
030520          J1=J+1
030530          IF(W1(I,J).EQ.0.)GOTO 702
030540          EXPY(J1)=ACEPT+RBERT1(I)*Y(I,J)
030550          IF(W1(I,J1).EQ.0.)GOTO 702
030560          DIF(J1)=Y(I,J1)-EXPY(J1)
030570          WRITE(8,606)J1,Y(I,J1),EXPY(J1),DIF(J1)
030580   606    FORMAT(1X,I3,T10,F14.4,T30,F14.4,T50,F14.4)
030590   702 CONTINUE
030600       WRITE(8,600)
030610    600 FORMAT('1')
030620       RETURN
030630       END
030640 C
030650 C*****************************************************************************
030660 C
030670       SUBROUTINE TRY2(I)
030680 CC
030690 CC----DESCRIPTION:  DERIVES AN INITIAL VALUE OF 'RHO' USING THE FIRST
030700 CC                  AND MIDDLE LENGTH GROUPS.
030710 CC
030720 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
030730 CC                        WYCOVE SYSTEMS LIMITED,
030740 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
030750 CC                        RESEARCH BRANCH, DFO, HALIFAX.
030760 CC
030770 CC----DATE:  MARCH 1984.
030780 CC
030790 CC----REVISION HISTORY:  NONE.
030800 CC
030810 CC----FILES:
030820 CC
030830 CC    UNIT    I/O    DESCRIPTION
030840 CC    ------------------------
030850 CC      8      O     RESULTS.
030860 CC
030870 CC----FORMAL PARAMETERS:
030880 CC
030890 CC    PARAMETER  I/O  DESCRIPTION
030900 CC    -------------------------
030910 CC    I           I   SAMPLE NUMBER.
030920 CC
030930 CC----COMMON BLOCKS:
030940 CC
030950 CC    COMMON /AA/
030960 CC
```

```
030970 CC    VARIABLE  DESCRIPTION
030980 CC    --------------------
030990 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
031000 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
031010 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
031020 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
031030 CC    N2        N + 1.
031040 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
031050 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
031060 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
031070 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
031080 CC
031090 CC    COMMON /ONER/
031100 CC
031110 CC    VARIABLE  DESCRIPTION
031120 CC    --------------------
031130 CC    IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
031140 CC              ESTIMATE OF 'RHO'.
031150 CC    TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
031160 CC    TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
031170 CC              DATA.
031180 CC    Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
031190 CC              AGE + 1.
031200 CC    DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
031210 CC    N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
031220 CC    SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
031230 CC    HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
031240 CC    IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
031250 CC              VALUE FOR 'B1/B'.
031260 CC    FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
031270 CC    IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
031280 CC              OF YEAR SAMPLE COLLECTED.
031290 CC    NWT       TYPE OF WEIGHTING:
031300 CC              1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
031310 CC              2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
031320 CC              3 - CONSTANT = 1.
031330 CC    NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
031340 CC              IF 'NTRY'=1 USE 'TRY1', ETC.
031350 CC
031360 CC    COMMON /AA1/
031370 CC
031380 CC    VARIABLE  DESCRIPTION
031390 CC    --------------------
031400 CC    RBERT1(10)    INITIAL ESTIMATE OF 'RHO' FROM 'TRY1' (2 OR 3).
031410 CC
031420 CC    COMMON /AA2/
031430 CC
031440 CC    VARIABLE  DESCRIPTION
031450 CC    --------------------
031460 CC    ITRY      CONTROLS HEADINGS IN 'TRY' ROUTINES.
031470 CC    AYINF     IN 'TRY2' AND 'TRY3', SET EQUAL TO NUMBER OF TIMES
031480 CC              ESTIMATE OF 'YINF' HAS TO BE MULTIPLIED BY 'FACK()'
031490 CC              TO GET A REASONABLE VALUE OF 'RHO'.
031500 CC
031510 CC    COMMON /AA3/
031520 CC
031530 CC    VARIABLE  DESCRIPTION
031540 CC    --------------------
031550 CC    YINF1     INITIAL ESTIMATE OF 'YINF' SET IN 'MISRA5'.
031560 CC
031570 CC    COMMON /AA10/
031580 CC
031590 CC    VARIABLE  DESCRIPTION
031600 CC    --------------------
031610 CC    FACK(10)  FACTOR FOR INCREASING ESTIMATE OF 'YINF' IN 'TRY2' OR
031620 CC              'TRY3'.
031630 CC    KN1       MAXIMUM NUMBER OF ITERATIONS OF LOOP THAT RE-ESTIMATES
031640 CC              'YINF' BY 'TRY' ROUTINES, THEN REDOES REGRESSION.
031650 CC
031660       COMMON /AA/   ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
031670       COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
031680      >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
```

```
031690       COMMON /AA1/  RBERT1(10)
031700       COMMON /AA2/  ITRY,AYINF
031710       COMMON /AA3/  YINF1
031720       COMMON /AA10/ FACK(10),KN1
031730 C
031740       ITRY=ITRY+1
031750       IF(ITRY.EQ.1) WRITE(8,5423)
031760  5423 FORMAT(1X,'USING TRY2')
031770 C
031780 C
031790 C-----ADJUST 'YINF' ESTIMATE.  FIND FIRST AND MIDDLE LENGTH WITH
031800 C     NON-ZERO SAMPLE.
031810       DO 4000 ITR=1,9999
031820           N6=N-ISTART+1
031830           YINF=YINF1
031840           YINF=YINF*AYINF
031850           IN=ISTART+N6/2
031860           DO 806 J=1,N6/2
031870               IF (Y(I,IN).NE.0.) GOTO 805
031880               IN=IN-1
031890   806     CONTINUE
031900   805     IN1=ISTART
031910           DO 807 J=1,N6/2
031920               IF (Y(I,IN1).NE.0.) GOTO 808
031930               IN1=IN1+1
031940   807     CONTINUE
031950   808     RBERT1(I)=((YINF-Y(I,IN))/(YINF-Y(I,IN1)))**(1./FLOAT(IN-IN1))
031960           AYINF=AYINF*FACK(I)
031970           IF (RBERT1(I).LT.1.0) GOTO 4072
031980  4000 CONTINUE
031990 4072  RETURN
032000       END
032010 C
032020 C*******************************************************************************
032030 C
032040       SUBROUTINE TRY3(I)
032050 CC
032060 CC----DESCRIPTION:  DERIVES AN INITIAL VALUE OF 'RHO' USING THE
032070 CC                  REGRESSION OF 'LOG(YINF-Y(T))' ON 'T'.
032080 CC
032090 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
032100 CC                         WYCOVE SYSTEMS LIMITED,
032110 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
032120 CC                         RESEARCH BRANCH, DFO, HALIFAX.
032130 CC
032140 CC----DATE:  MARCH 1984.
032150 CC
032160 CC----REVISION HISTORY:  NONE.
032170 CC
032180 CC----FILES:
032190 CC
032200 CC    UNIT    I/O    DESCRIPTION
032210 CC    ------------------------
032220 CC      8      O     RESULTS.
032230 CC
032240 CC----FORMAL PARAMETERS:
032250 CC
032260 CC    PARAMETER  I/O  DESCRIPTION
032270 CC    -------------------------
032280 CC    I           I   SAMPLE NUMBER.
032290 CC
032300 CC----COMMON BLOCKS:
032310 CC
032320 CC    COMMON /AA/
032330 CC
032340 CC    VARIABLE  DESCRIPTION
032350 CC    --------------------
032360 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
032370 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
032380 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
032390 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
032400 CC    N2        N + 1.
```

```
032410 CC     W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
032420 CC     W3(10,30) TEMPORARY COPY OF 'W1(,)'.
032430 CC     LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
032440 CC     LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
032450 CC
032460 CC     COMMON /ONER/
032470 CC
032480 CC     VARIABLE  DESCRIPTION
032490 CC     --------------------
032500 CC     IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
032510 CC               ESTIMATE OF 'RHO'.
032520 CC     TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
032530 CC     TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
032540 CC               DATA.
032550 CC     Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
032560 CC               AGE + 1.
032570 CC     DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
032580 CC     N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
032590 CC     SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
032600 CC     HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
032610 CC     IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
032620 CC               VALUE FOR 'B1/B'.
032630 CC     FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
032640 CC     IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
032650 CC               OF YEAR SAMPLE COLLECTED.
032660 CC     NWT       TYPE OF WEIGHTING:
032670 CC               1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
032680 CC               2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
032690 CC               3 - CONSTANT = 1.
032700 CC     NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
032710 CC               IF 'NTRY'=1 USE 'TRY1', ETC.
032720 CC
032730 CC     COMMON /AA1/
032740 CC
032750 CC     VARIABLE  DESCRIPTION
032760 CC     --------------------
032770 CC     RBERT1(10)   INITIAL ESTIMATE OF 'RHO' FROM 'TRY1' (2 OR 3).
032780 CC
032790 CC     COMMON /AA2/
032800 CC
032810 CC     VARIABLE  DESCRIPTION
032820 CC     --------------------
032830 CC     ITRY      CONTROLS HEADINGS IN 'TRY' ROUTINES.
032840 CC     AYINF     IN 'TRY2' AND 'TRY3', SET EQUAL TO NUMBER OF TIMES
032850 CC               ESTIMATE OF 'YINF' HAS TO BE MULTIPLIED BY 'FACK()'
032860 CC               TO GET A REASONABLE VALUE OF 'RHO'.
032870 CC
032880 CC     COMMON /AA3/
032890 CC
032900 CC     VARIABLE  DESCRIPTION
032910 CC     --------------------
032920 CC     YINF1     INITIAL ESTIMATE OF 'YINF' SET IN 'MISRA5'.
032930 CC
032940 CC     COMMON /AA10/
032950 CC
032960 CC     VARIABLE  DESCRIPTION
032970 CC     --------------------
032980 CC     FACK(10)  FACTOR FOR INCREASING ESTIMATE OF 'YINF' IN 'TRY2' OR
032990 CC               'TRY3'.
033000 CC     KN1       MAXIMUM NUMBER OF ITERATIONS OF LOOP THAT RE-ESTIMATES
033010 CC               'YINF' BY 'TRY' ROUTINES, THEN REDOES REGRESSION.
033020 CC
033030       COMMON /AA/   ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
033040       COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
033050      >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
033060       COMMON /AA1/  RBERT1(10)
033070       COMMON /AA2/  ITRY,AYINF
033080       COMMON /AA3/  YINF1
033090       COMMON /AA10/ FACK(10),KN1
033100 C
033110       DIMENSION Y1(10,30)
033120 C
```

```
033130       ITRY=ITRY+1
033140       IF (ITRY.EQ.1) WRITE(8,5423)
033150  5423 FORMAT(1X,'USING TRY3')
033160 C
033170 C
033180 C-----ADJUST 'YINF' ESTIMATE.  REGRESS 'LOG(YINF-Y(T))' ON 'T'.
033190       DO 4000 ITR=1,9999
033200           YINF=YINF1
033210           YINF=YINF*AYINF
033220           SW=0.
033230           SWX=0.
033240           SWY=0.
033250           SWXY=0.
033260           SWXX=0.
033270           DO 701 K=ISTART,N
033280               IF (Y(I,K).NE.0.) THEN
033290                   Y1(I,K)=ALOG10(YINF-Y(I,K))
033300                   SW=SW+W1(I,K)
033310                   SWX=SWX+W1(I,K)*K
033320                   SWXX=SWXX+W1(I,K)*K**2
033330                   SWY=SWY+W1(I,K)*Y1(I,K)
033340                   SWXY=SWXY+W1(I,K)*Y1(I,K)*K
033350               ENDIF
033360   701     CONTINUE
033370           RBERT1(I)=10.**((SWXY-SWX*SWY/SW)/(SWXX-SWX**2/SW))
033380           AYINF=AYINF*FACK(I)
033390           IF(RBERT1(I).LT.1.0)GOTO 4072
033400  4000 CONTINUE
033410  4072 RETURN
033420       END
033430 C
033440 C*****************************************************************************
033450 C
033460       SUBROUTINE MISRA4
033470 CC
033480 CC----DESCRIPTION:  GENERATES AND OUTPUTS CONFIDENCE LIMITS BASED ON
033490 CC                  POOLED MEAN SQUARE AND CHI-SQUARE AMONG SAMPLES.
033500 CC
033510 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
033520 CC                          WYCOVE SYSTEMS LIMITED,
033530 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
033540 CC                          RESEARCH BRANCH, DFO, HALIFAX.
033550 CC
033560 CC----DATE:  MARCH 1984.
033570 CC
033580 CC----REVISION HISTORY:  NONE.
033590 CC
033600 CC----FILES:
033610 CC
033620 CC    UNIT    I/O    DESCRIPTION
033630 CC    ------------------------
033640 CC      8      O     RESULTS.
033650 CC
033660 CC----COMMON BLOCKS:
033670 CC
033680 CC    COMMON /ONER/
033690 CC
033700 CC    VARIABLE  DESCRIPTION
033710 CC    --------------------
033720 CC    IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
033730 CC              ESTIMATE OF 'RHO'.
033740 CC    TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
033750 CC    TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
033760 CC              DATA.
033770 CC    Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
033780 CC              AGE + 1.
033790 CC    DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
033800 CC    N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
033810 CC    SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
033820 CC    HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
033830 CC    IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
033840 CC              VALUE FOR 'B1/B'.
```

```
033850 CC     FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
033860 CC     IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
033870 CC               OF YEAR SAMPLE COLLECTED.
033880 CC     NWT       TYPE OF WEIGHTING:
033890 CC               1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
033900 CC               2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
033910 CC               3 - CONSTANT = 1.
033920 CC     NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
033930 CC               IF 'NTRY'=1 USE 'TRY1', ETC.
033940 CC
033950 CC     COMMON /FNX/
033960 CC
033970 CC     VARIABLE  DESCRIPTION
033980 CC     ----------------------
033990 CC     INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
034000 CC     INX1      INX + 1.
034010 CC
034020 CC     COMMON /FNY/
034030 CC
034040 CC     VARIABLE  DESCRIPTION
034050 CC     ----------------------
034060 CC     INY       NUMBER OF SAMPLES.
034070 CC     INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
034080 CC
034090 CC     COMMON /COM/
034100 CC
034110 CC     VARIABLE  DESCRIPTION
034120 CC     ----------------------
034130 CC     SCASSE    VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
034140 CC               ANOVA.
034150 CC     ISCAE     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
034160 CC               ANOVA.
034170 CC     SCASS     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
034180 CC               ANOVA.
034190 CC     ISCA      VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
034200 CC               ANOVA.
034210 CC     SUMRSS    VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
034220 CC               ANOVA.
034230 CC     COMSS     VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
034240 CC               ANOVA.
034250 CC     SR(2,10)  VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
034260 CC               ANOVA.
034270 CC     SR1(10)   VALUE SAVED FROM FIT TO COMBINED DATA TO BE USED IN
034280 CC               ANOVA.
034290 CC
034300 CC     COMMON /CCM/
034310 CC
034320 CC     VARIABLE  DESCRIPTION
034330 CC     ----------------------
034340 CC     PARM(10,3)   FIT PARAMETERS FOR EACH SAMPLE:  'YINF', 'B', 'RHO'.
034350 CC     VPARM(10,3)  VARIANCE ESTIMATES FOR FIT PARAMETERS.
034360 CC
034370       COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
034380      >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
034390       COMMON /FNX/  INX,INX1
034400       COMMON /FNY/  INY,INY1
034410       COMMON /COM/  SCASSE,ISCAE,SCASS,ISCA,SUMRSS,COMSS,SR(2,10),
034420      >              SR1(10)
034430       COMMON /CMM/  PARM(10,3),VPARM(10,3)
034440 C
034450       DIMENSION SPARM(10,3),PARML(10,3),PARMU(10,3),WI(10,3),SUMP(10),
034460      >          SUMW(10),DIFFP(10,3),CHI(10),AVGP(10)
034470 C
034480       SCAMSE=SCASSE/ISCAE
034490       DO 733 I=1,INY1
034500         DO 733 J=1,INX1
034510           SPARM(I,J)=SQRT(VPARM(I,J)*SCAMSE)
034520           WI(I,J)=1./(VPARM(I,J)*SCAMSE)
034530           PARML(I,J)=PARM(I,J)-TT*SPARM(I,J)
034540           PARMU(I,J)=PARM(I,J)+TT*SPARM(I,J)
034550   733 CONTINUE
034560       DO 702 J=1,INX1
```

```
034570            SUMP(J)=0.
034580            SUMW(J)=0.
034590            DO 703 I=1,INY
034600                SUMP(J)=SUMP(J)+WI(I,J)*PARM(I,J)
034610                SUMW(J)=SUMW(J)+WI(I,J)
034620  703       CONTINUE
034630  702   AVGP(J)=SUMP(J)/SUMW(J)
034640        DO 704 J=1,INX1
034650            CHI(J)=0.
034660          DO 704 I=1,INY
034670              DIFFP(I,J)=PARM(I,J)-AVGP(J)
034680              CHI(J)=CHI(J)+WI(I,J)*DIFFP(I,J)**2
034690  704   CONTINUE
034700        IDF=INY-1
034710  601   FORMAT(/)
034720  602   FORMAT(//)
034730        WRITE(8,603)
034740  603   FORMAT(1X,'CONSTANT',T26,'VALUE',T41,'SE',T56,'CONFIDENCE    ',
034750       >       ' LIMIT'/T55,'LOWER',T70,'UPPER')
034760        WRITE(8,601)
034770        DO 701 I=1,INY
034780            WRITE(8,601)
034790            WRITE(8,631)I
034800  631       FORMAT(25X,'FOR SAMPLE ',I2,2X,'BASED ON POOLED ERROR MS')
034810            WRITE(8,601)
034820            WRITE(8,604)(PARM(I,J),SPARM(I,J),PARML(I,J),PARMU(I,J)
034830       >                ,J=1,INX1)
034840  604       FORMAT(1X,'Y-INFINITY',T19,F14.6,T34,F14.6,T49,F14.6,T62,
034850       >    F14.6/1X,'SLOPE',T19,F14.6,T34,F14.6,T49,F14.6,T62,F14.6/1X,
034860       >    'RHO',T19,F14.6,T34,F14.6,T49,F14.6,T62,F14.6)
034870  701   CONTINUE
034880        WRITE(8,601)
034890        WRITE(8,607)IDF
034900  607   FORMAT(1X,'CHI-SQUARE VALUES, EACH ON D.F.= ',I3,/1X,
034910       >      'FOR DIFFERENCES AMONG SAMPLES IN :')
034920        WRITE(8,601)
034930        WRITE(8,608)(CHI(J),J=1,INX1)
034940  608   FORMAT(1X,'Y-INFINITY',T30,F14.4/1X,'SLOPES (B) ',
034950       >      T30,F14.4/1X,'RHO ',T30,F14.4)
034960        WRITE(8,601)
034970        RETURN
034980        END
034990 C
035000 C*************************************************************************
035010 C
035020        SUBROUTINE MISRA5(I)
035030 CC
035040 CC----DESCRIPTION:  CALCULATES INITIAL ESTIMATE OF 'YINF'.
035050 CC
035060 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
035070 CC                        WYCOVE SYSTEMS LIMITED,
035080 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
035090 CC                        RESEARCH BRANCH, DFO, HALIFAX.
035100 CC
035110 CC----DATE:  MARCH 1984.
035120 CC
035130 CC----REVISION HISTORY:  NONE.
035140 CC
035150 CC----FORMAL PARAMETERS:
035160 CC
035170 CC    PARAMETER   I/O   DESCRIPTION
035180 CC    ---------------------------
035190 CC    I            I    SAMPLE NUMBER.
035200 CC
035210 CC----COMMON BLOCKS:
035220 CC
035230 CC    COMMON /AA/
035240 CC
035250 CC    VARIABLE  DESCRIPTION
035260 CC    --------------------
035270 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
035280 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
```

```
035290 CC     N          ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
035300 CC                THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
035310 CC     N2         N + 1.
035320 CC     W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
035330 CC     W3(10,30) TEMPORARY COPY OF 'W1(,)'.
035340 CC     LNA(30)    NUMBER OF SAMPLES WITH NO DATA, BY AGE.
035350 CC     LNY(10)    NUMBER OF AGES WITH NO DATA, BY SAMPLE.
035360 CC
035370 CC     COMMON /ONER/
035380 CC
035390 CC     VARIABLE  DESCRIPTION
035400 CC     --------------------
035410 CC     IN2(10)    MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
035420 CC                ESTIMATE OF 'RHO'.
035430 CC     TT         NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
035440 CC     TT1(10)    NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
035450 CC                DATA.
035460 CC     Y(10,30)   MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
035470 CC                AGE + 1.
035480 CC     DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
035490 CC     N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
035500 CC     SUMRS      SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
035510 CC     HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
035520 CC     IN3(10)    MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
035530 CC                VALUE FOR 'B1/B'.
035540 CC     FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
035550 CC     IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
035560 CC                OF YEAR SAMPLE COLLECTED.
035570 CC     NWT        TYPE OF WEIGHTING:
035580 CC                1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
035590 CC                2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
035600 CC                3 - CONSTANT = 1.
035610 CC     NTRY       SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
035620 CC                IF 'NTRY'=1 USE 'TRY1', ETC.
035630 CC
035640 CC     COMMON /AA3/
035650 CC
035660 CC     VARIABLE  DESCRIPTION
035670 CC     --------------------
035680 CC     YINF1      INITIAL ESTIMATE OF 'YINF' SET IN 'MISRA5'.
035690 CC
035700 CC     COMMON /AA12/
035710 CC
035720 CC     VARIABLE  DESCRIPTION
035730 CC     --------------------
035740 CC     FACK1      FACTOR USED TO INCREMENT ESTIMATE OF 'YINF' IN 'MISRA5'.
035750 CC
035760       COMMON /AA/   ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
035770       COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
035780      >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
035790       COMMON /AA3/  YINF1
035800       COMMON /AA12/ FACK1
035810 C
035820 C
035830 C-----FIND LAST AGE GROUP WITH POSITIVE MEAN LENGTH.
035840       N6=N
035850       AY=1.
035860       DO 802 J=1,N-1
035870           IF (Y(I,N6).NE.0.) GOTO 801
035880           N6=N6-1
035890   802 CONTINUE
035900   801 DO 803 IY=1,9999
035910           AY=AY*FACK1
035920           YINF1=Y(I,N6)*AY
035930 C
035940 C
035950 C-----CHECK THAT 'YINF' ESTIMATE IS LARGER THAN ALL MEAN LENGTHS, ELSE
035960 C      MAKE IT A LITTLE BIGGER.
035970           DO 703 K=ISTART,N6
035980               IF (YINF1.LE.Y(I,K)) GOTO 803
035990   703     CONTINUE
036000           GOTO 804
```

```
036010  803  CONTINUE
036020  804  RETURN
036030       END
036040 C
036050 C*************************************************************************
036060 C
036070       SUBROUTINE MISRA6(A,B,C,D)
036080 CC
036090 CC----DESCRIPTION:  GENERATES 'YINF', 'K', AND 'T0' FOR DATA TRANSFORMED
036100 CC                  BY 'D' AND UNTRANSFORMED FROM PARAMETERS FROM THE
036110 CC                  GENERALIZED VON BERTALANFFY FIT.
036120 CC
036130 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
036140 CC                        WYCOVE SYSTEMS LIMITED,
036150 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
036160 CC                        RESEARCH BRANCH, DFO, HALIFAX.
036170 CC
036180 CC----DATE:  MARCH 1984.
036190 CC
036200 CC----REVISION HISTORY:  NONE.
036210 CC
036220 CC----FILES:
036230 CC
036240 CC    UNIT    I/O    DESCRIPTION
036250 CC    ------------------------
036260 CC      8      O     RESULTS.
036270 CC
036280 CC----FORMAL PARAMETERS:
036290 CC
036300 CC    PARAMETER  I/O  DESCRIPTION
036310 CC    --------------------------
036320 CC    A           I   ESTIMATE OF 'YINF'.
036330 CC    B           I   ESTIMATE OF 'BETA'.
036340 CC    C           I   ESTIMATE OF 'RHO'.
036350 CC    D           I   ESTIMATE OF 'D'.
036360 CC
036370  601  FORMAT(/)
036400       A3=A
036410       A1=-ALOG(C)
036420       A2=(ALOG(-A/B))/ALOG(C)
036430       WRITE(8,604)
036440  604  FORMAT(1X,'FOR THE TRANSFORMED DATA')
036520       WRITE(8,601)
036530       WRITE(8,603)A3,A1,A2
036540  603  FORMAT(1X,'VALUES DERIVED FROM '
036550      >          ,'EQUATION (1) :'//1X,'Y-INFINITY =',T20,F14.6/1X,
036560      >          'K =',T20,F14.6/1X,'TIME-ZERO =',T20,F14.6)
036590       WRITE(8,601)
036600       RETURN
036610       END
036620 C
036630 C*************************************************************************
036640 C
036650       SUBROUTINE MISRA7(I)
036660 CC
036670 CC----DESCRIPTION:  RE-ESTIMATES PARAMETERS BY ALLEN'S METHOD.
036680 CC
036690 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
036700 CC                        WYCOVE SYSTEMS LIMITED,
036710 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
036720 CC                        RESEARCH BRANCH, DFO, HALIFAX.
036730 CC
036740 CC----DATE:  MARCH 1984.
036750 CC
036760 CC----REVISION HISTORY:  NONE.
036770 CC
036780 CC----FILES:
036790 CC
036800 CC    UNIT    I/O    DESCRIPTION
036810 CC    ------------------------
036820 CC      8      O     RESULTS.
036830 CC
```

```
.036840 CC----FORMAL PARAMETERS:
 036850 CC
 036860 CC    PARAMETER  I/O  DESCRIPTION
 036870 CC    -------------------------
 036880 CC    I           I   SAMPLE NUMBER.
 036890 CC
 036900 CC
 036910 CC----EXTERNALS:
 036920 CC
 036930 CC    FUNCTIONS  FNX1, FNX2, FNY1, FNY2, FNXY.
 036940 CC
 036950 CC----COMMON BLOCKS:
 036960 CC
 036970 CC    COMMON /AA/
 036980 CC
 036990 CC    VARIABLE  DESCRIPTION
 037000 CC    --------------------
 037010 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
 037020 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
 037030 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
 037040 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
 037050 CC    N2        N + 1.
 037060 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
 037070 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
 037080 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
 037090 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
 037100 CC
 037110 CC    COMMON /ONER/
 037120 CC
 037130 CC    VARIABLE  DESCRIPTION
 037140 CC    --------------------
 037150 CC    IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
 037160 CC              ESTIMATE OF 'RHO'.
 037170 CC    TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
 037180 CC    TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
 037190 CC              DATA.
 037200 CC    Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
 037210 CC              AGE + 1.
 037220 CC    DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
 037230 CC    N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
 037240 CC    SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
 037250 CC    HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
 037260 CC    IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
 037270 CC              VALUE FOR 'B1/B'.
 037280 CC    FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
 037290 CC    IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
 037300 CC              OF YEAR SAMPLE COLLECTED.
 037310 CC    NWT       TYPE OF WEIGHTING:
 037320 CC              1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
 037330 CC              2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
 037340 CC              3 - CONSTANT = 1.
 037350 CC    NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
 037360 CC              IF 'NTRY'=1 USE 'TRY1', ETC.
 037370 CC
 037380 CC    COMMON /FNX/
 037390 CC
 037400 CC    VARIABLE  DESCRIPTION
 037410 CC    --------------------
 037420 CC    INX       NUMBER OF INDEPENDENT VARIABLES IN REGRESSION.
 037430 CC    INX1      INX + 1.
 037440 CC
 037450 CC    COMMON /FNY/
 037460 CC
 037470 CC    VARIABLE  DESCRIPTION
 037480 CC    --------------------
 037490 CC    INY       NUMBER OF SAMPLES.
 037500 CC    INY1      INY + 1, EXTRA COLUMN FOR TOTAL OVER SAMPLES.
 037510 CC
 037520 CC    COMMON /AA4/
 037530 CC
 037540 CC    VARIABLE  DESCRIPTION
 037550 CC    --------------------
```

```
037560 CC    RBERT2(10)   FINAL ESTIMATE OF 'RHO' FROM MISRA1.
037570 CC
037580 CC    COMMON /AA11/
037590 CC
037600 CC    VARIABLE  DESCRIPTION
037610 CC    --------------------
037620 CC    DFAC(10)  'D' VALUE FOR SAMPLE.
037630 CC    DMAX      STARTING (MAXIMUM) VALUE FOR 'D'.
037640 CC    STEP      VALUE SUBTRACTED FROM 'D' EACH ITERATION.
037650 CC    DMIN      MINIMUM VALUE FOR 'D'.
037660 CC    Y4(10,30) TEMPORARY STORAGE FOR 'Y(,)'.
037670 CC
037680 CC    COMMON /AA20/
037690 CC
037700 CC    VARIABLE  DESCRIPTION
037710 CC    --------------------
037720 CC    DFACL     BOUNDS USED TO AVOID EXPONENTS CLOSE TO 1.
037730 CC    DFACU     BOUNDS USED TO AVOID EXPONENTS CLOSE TO 1.
037740 CC
037750 CC----STATEMENT FUNCTIONS:
037760 CC
037770 CC    PROTOTYPE  DESCRIPTION
037780 CC    --------------------
037790 CC    SN4(I)      EXP(-I*ESTIMATE OF 'K').
037800 CC    SN5(I)      I*SN4(I).
037810 CC    SN6(I)      I**2*SN4(I).
037820 CC
037830       SN4(K)=EXP(-BERTK*K)
037840       SN5(K)=K*SN4(K)
037850       SN6(K)=K**2*SN4(K)
037860 C
037870       COMMON /AA/    ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
037880       COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
037890      >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
037900       COMMON /FNX/  INX,INX1
037910       COMMON /FNY/  INY,INY1
037920       COMMON /AA4/  RBERT2(10)
037930       COMMON /AA11/ DFAC(10),DMAX,STEP,DMIN,Y4(10,30)
037940       COMMON /AA20/ DFACL,DFACU
037950 C
037960       DIMENSION X(3,30)
037970 C
037980 C
037990 C-----SET PARAMETERS.  WE ARE WORKING WITH 'K' (-LOG('RHO')) IN THIS
038000 C     METHOD.  USE SOLUTIONS FROM 'MISRA1' TO START THE ITERATIONS.
038010       INN2=10
038020       INN3=100
038030       DD=DFAC(I)
038040       A1A=0.
038050       BERTK=-ALOG(RBERT2(I))
038060  601  FORMAT(/)
038070       AKOUNT=0.
038080       DO 3050 J=ISTART,N
038090           IF (W1(I,J).GT.0.) AKOUNT=AKOUNT+1
038100  3050 CONTINUE
038110       DO 701 J1=1,N2
038120          DO 701 J2=1,INX1
038130             X(J2,J1)=0.
038140   701 CONTINUE
038150       A=0.
038160       DO 702 J=ISTART,N
038170           A=A+W1(I,J)
038180   702 CONTINUE
038190       B=FNY1(I,Y)
038200 C
038210 C
038220 C-----LOOP TO CONTROL PRECISION OF CORRECTION CUTOFF.  USES SAME
038230 C     INCREMENT AS IN 'MISRA1'.
038240       DO 980 I4=1,INN3
038250          A1A=A1A+FACK4(I)
038260 C
038270 C
```

```
038280 C-----LOOP TO IMPROVE ESTIMATE OF 'K'.  CUT OFF IF CORRECTION LESS THAN
038290 C     TOLERANCE SET IN 980 LOOP.
038300           DO 970 IIN2=1,INN2
038310               DO 788 I1=ISTART,N
038320                   I2=I1-1
038330                   X(1,I1)=SN4(I2)
038340                   X(2,I1)=SN5(I2)
038350                   X(3,I1)=SN6(I2)
038360  788          CONTINUE
038370               C=FNX1(1,X,I)
038380               D=FNXY(1,I,X,Y)
038390               E=FNX2(1,1,X,I)
038400               F=FNXY(2,I,X,Y)
038410               KK=C
038420               G=FNX1(2,X,I)
038430               H=FNX2(1,2,X,I)
038440               AI=FNX2(2,2,X,I)
038450               AJ=FNX1(3,X,I)
038460               AL=FNXY(3,I,X,Y)
038470               AM=B*E-C*D
038480               P=A*D-B*C
038490               Q=A*E-C**2.
038500               R=C*G-A*H
038510               S=B*G-A*F
038520               T=2.*A*AI-G*G-C*AJ
038530               U=A*AL-B*AJ
038540               V=G*D+C*F-2.*B*H
038550               ALPHA=Q*S-P*R
038560               DKAL=Q*U+R*S-P*T
038570               CORREC=-ALPHA/DKAL
038580               A1B=ABS(CORREC/BERTK)
038590               PREVK=BERTK
038600               BERTK=PREVK+CORREC
038610 C
038620 C
038630 C-----WE HAVE A CORRECTION BELOW TOLERANCE.  LET'S SEE IF WE WANT THE
038640 C     SOLUTION.
038650               IF (A1B.LE.A1A) THEN
038660                   TZERON=ALOG(-P)-ALOG(AM)
038670                   TZERO=(TZERON+CORREC*(S*AM-P*V)/(P*AM))/BERTK
038680                   TZERON=TZERON/BERTK
038690                   YINF=AM/Q
038700                   YINFC=YINF+(BERTK-PREVK)*((Q*V-2.*AM*R)/(Q**2.))
038710                   IF(DD.GT.DFACL.AND.DD.LT.DFACU) THEN
038720                       AAY=1./DFAC(I)
038730                       FINALY=YINFC**AAY
038740                       FINALK=BERTK*AAY
038750                     ELSE
038760                       FINALY=YINFC
038770                       FINALK=BERTK
038780                   ENDIF
038790                   Z=-P/AM
038800                   A1=A+Z*Z*E-2.*Z*C
038810                   A2=YINF*PREVK*Z*(Z*E-C)
038820                   A3=YINF*Z*Z*(TZERON*E-H)+YINF*Z*(G-TZERON*C)
038830                   B2=YINF*YINF*PREVK*PREVK*Z*Z*E
038840                   C2=YINF*YINF*PREVK*Z*Z*(TZERON*E-H)
038850                   C3=YINF*YINF*Z*Z*(TZERON*TZERON*E-2.*TZERON*H+AI)
038860                   B1=A2
038870                   C1=A3
038880                   B3=C2
038890                   ERRSS=FNY2(I,Y)-B*B/A-P*P/(A*Q)
038900                   ERRDF=AKOUNT-3
038910                   IERRDF=ERRDF
038920                   ERRMS=ERRSS/ERRDF
038930                   DEL=A1*B2*C3+2.*A2*C2*A3-A1*C2*C2-B2*A3*A3-C3*A2*A2
038940                   DEL=ERRMS/DEL
038950 C-----CHECK IF SOLUTION OK.
038960                   IF (DEL.LT.0.) GOTO 980
038970                   IF (B2*C3.LT.C2*C2) GOTO 980
038980                   IF (C3*A1.LT.A3*A3) GOTO 980
038990                   IF (A1*B2.LT.A2*A2) GOTO 980
```

```
039000                  GOTO 874
039010                ENDIF
039020  970        CONTINUE
039030  980   CONTINUE
039040 C
039050 C
039060 C-----IF WE GET TO HERE, WE DID NOT FIND A GOOD FIT.  JUMP TO 872.
039070        GOTO 872
039080 C
039090 C
039100 C-----WE HAVE A GOOD SOLUTION, SO CALCULATE S.E.S AND PRINT RESULTS.
039110  874   SYINFC=SQRT((B2*C3-C2*C2)*DEL)
039120        STZERO=SQRT((C3*A1-A3*A3)*DEL)
039130        SBERTK=SQRT((A1*B2-A2*A2)*DEL)
039140        IF(DD.GT.DFACL.AND.DD.LT.DFACU) THEN
039150            SBERTK=SBERTK/DD
039160            AAY=(1.-DD)/DD
039170            SYINFC=((YINFC**AAY)/DD)*SYINFC
039180        ENDIF
039190        WRITE(8,601)
039200        WRITE(8,604)
039210  604   FORMAT(1X,'  BY ALLENS ALGORITHM:')
039220        WRITE(8,601)
039230        A1A=100.*A1A
039240        WRITE(8,601)
039250        WRITE(8,601)
039260        WRITE(8,605)
039270  605   FORMAT(10X,' CONSTANT',T44,'SE')
039280        WRITE(8,601)
039290        WRITE(8,606) FINALK,SBERTK,TZERO,STZERO,FINALY,SYINFC
039300  606   FORMAT(3X,'K= ',T16,F14.6,T38,F14.6/3X,'T-ZERO= ',T16,F14.6,
039310       >       T38,F14.6/3X,'Y-INFINITY= ',T16,F14.6,T38,F14.6)
039320        WRITE(8,601)
039330        RETURN
039340 C
039350 C
039360 C-----COMES HERE IF DID NOT FIND FIT.
039370  872   WRITE(8,603) INN3
039380  603   FORMAT(1X,'FOR ALLENS ALGORITM WATCH IT - IT IS ',I4,
039390       >       ' ITERATIONS NOW')
039400        WRITE(8,601)
039410        RETURN
039420        END
039430 C
039440 C*****************************************************************************
039450 C
039460        SUBROUTINE MISRA9(K,X,Y3,NK)
039470 CC
039480 CC----DESCRIPTION:  FITS PARABOLA TO THE LOG OF THE MEAN SQUARES OF THE
039490 CC                  SAMPLE DATA, THEN CALCULATES WEIGHTS AS
039500 CC                  FREQ./ANTILOG(.5*PREDICTED MS).
039510 CC
039520 CC----ORIGIN:  MODIFIED BY DAVID GRAY,
039530 CC                        WYCOVE SYSTEMS LIMITED,
039540 CC             FROM CODE SUPPLIED BY DR. R.K. MISRA,
039550 CC                        RESEARCH BRANCH, DFO, HALIFAX.
039560 CC
039570 CC----DATE:  MARCH 1984.
039580 CC
039590 CC----REVISION HISTORY:  NONE.
039600 CC
039610 CC----FILES:
039620 CC
039630 CC    UNIT    I/O    DESCRIPTION
039640 CC    ------------------------
039650 CC      8      O     RESULTS.
039660 CC
039670 CC----FORMAL PARAMETERS:
039680 CC
039690 CC    PARAMETER  I/O  DESCRIPTION
039700 CC    -------------------------
039710 CC    K           I   SAMPLE NUMBER.
```

```
039720 CC    X           I   VECTOR OF AGES.
039730 CC    Y3          I   SAMPLE MS.
039740 CC    NK          I   OLDEST AGE.
039750 CC
039760 CC----COMMON BLOCKS:
039770 CC
039780 CC    COMMON /AA/
039790 CC
039800 CC    VARIABLE  DESCRIPTION
039810 CC    --------------------
039820 CC    ISTART    STARTING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
039830 CC              THAN YOUNGEST AGE (AGE 0 IN POSITION 1).
039840 CC    N         ENDING INDEX FOR AGE USED TO LOCATE DATA, ONE GREATER
039850 CC              THAN OLDEST AGE.  MAX VALUE IS 29, SO MAX AGE IS 28.
039860 CC    N2        N + 1.
039870 CC    W1(10,30) WEIGHTS FOR MEAN LENGTH AT AGE.
039880 CC    W3(10,30) TEMPORARY COPY OF 'W1(,)'.
039890 CC    LNA(30)   NUMBER OF SAMPLES WITH NO DATA, BY AGE.
039900 CC    LNY(10)   NUMBER OF AGES WITH NO DATA, BY SAMPLE.
039910 CC
039920 CC    COMMON /ONER/
039930 CC
039940 CC    VARIABLE  DESCRIPTION
039950 CC    --------------------
039960 CC    IN2(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP ADDING 'B1/B' TO
039970 CC              ESTIMATE OF 'RHO'.
039980 CC    TT        NUMBER OF S.E.S FOR CONF. LIMITS USING POOLED VALUES.
039990 CC    TT1(10)   NUMBER OF S.E.S FOR CONF. LIMITS USING INDIVIDUAL SAMPLE
040000 CC              DATA.
040010 CC    Y(10,30)  MEAN LENGTH AT AGE; (I,J) - I IS SAMPLE NUMBER, J IS
040020 CC              AGE + 1.
040030 CC    DIFF(10,30)   OBSERVED-PREDICTED LENGTH AT AGE.
040040 CC    N1(10,30) NUMBER OF DATA VALUES IN MEAN LENGTH.
040050 CC    SUMRS     SS ATTRIBUTABLE TO REGRESSION OVER ALL SAMPLES.
040060 CC    HARN1(10) HARMONIC MEAN NUMBER OF OBSERVATIONS.
040070 CC    IN3(10)   MAXIMUM NUMBER OF ITERATIONS, LOOP INCREASING CUTOFF
040080 CC              VALUE FOR 'B1/B'.
040090 CC    FACK4(10) FIRST CUTOFF VALUE FOR 'B1/B' AND INCREMENT.
040100 CC    IYEAR(10) INPUT CODE TO SELECT SAMPLE, USUALLY LAST TWO DIGITS
040110 CC              OF YEAR SAMPLE COLLECTED.
040120 CC    NWT       TYPE OF WEIGHTING:
040130 CC              1 - WEIGHTS DERIVED FROM FITTING PARABOLA TO MS DATA.
040140 CC              2 - WEIGHTS ARE SAMPLE FREQUENCY/POOLED MS.
040150 CC              3 - CONSTANT = 1.
040160 CC    NTRY      SUBROUTINE USED TO GET FIRST ESTIMATE OF 'RHO';
040170 CC              IF 'NTRY'=1 USE 'TRY1', ETC.
040180 CC
040190 CC    COMMON /AA9/
040200 CC
040210 CC    VARIABLE  DESCRIPTION
040220 CC    --------------------
040230 CC    VMS(10,30)   SAMPLE MEAN SQUARE; IN 'MISRA9' SET TO SQUARE ROOT OF
040240 CC                 PREDICTED SAMPLE MEAN SQUARE.
040250 CC
040260       COMMON /AA/   ISTART,N,N2,W1(10,30),W3(10,30),LNA(30),LNY(10)
040270       COMMON /ONER/ IN2(10),TT,TT1(10),Y(10,30),DIFF(10,30),N1(10,30),
040280      >              SUMRS,HARN1(10),IN3(10),FACK4(10),IYEAR(10),NWT,NTRY
040290       COMMON /AA9/  VMS(10,30)
040300 C
040310       DIMENSION X(NK),Y3(NK),B(3),Z(30)
040320 C
040330 C
040340 C-----INITIALIZE SUMS.
040350       SX1=0.
040360       SY1=0.
040370       SY2=0.
040380       SZ1=0.
040390       SZ2=0.
040400       SXZ=0.
040410       SXY=0.
040420       SZY=0.
040430       KOUNT=N-ISTART+1
```

```
040440  601  FORMAT(/)
040450 C
040460 C
040470 C-----NOTE:  THE LOG OF THE MEAN SQUARES IS BEING FITTED.
040480       DO 702 I=ISTART,N
040490           IF(Y3(I).GT.0.) Y3(I)=ALOG(Y3(I))
040500  702  CONTINUE
040510       DO 701 I=ISTART,N
040520           IF(N1(K,I).LE.1) THEN
040530               KOUNT=KOUNT-1
040540             ELSE
040550               Z(I)=X(I)**2
040560               SY1=SY1+Y3(I)
040570               SY2=SY2+Y3(I)**2
040580               SX1=SX1+X(I)
040590               SZ1=SZ1+Z(I)
040600               SXZ=SXZ+X(I)*Z(I)
040610               SXY=SXY+X(I)*Y3(I)
040620               SZ2=SZ2+Z(I)**2
040630               SZY=SZY+Z(I)*Y3(I)
040640           ENDIF
040650  701  CONTINUE
040660       SSY=SY2-SY1**2/KOUNT
040670       SSX=SZ1-SX1**2/KOUNT
040680       SPXZ=SXZ-SX1*SZ1/KOUNT
040690       SPXY=SXY-SX1*SY1/KOUNT
040700       SSZ=SZ2-SZ1**2/KOUNT
040710       SPZY=SZY-SZ1*SY1/KOUNT
040720       D=SSX*SSZ-SPXZ**2
040730       B(2)=(SSZ*SPXY-SPXZ*SPZY)/D
040740       B(3)=(SSX*SPZY-SPXZ*SPXY)/D
040750       YBAR=SY1/KOUNT
040760       XBAR=SX1/KOUNT
040770       ZBAR=SZ1/KOUNT
040780       B(1)=YBAR-B(2)*XBAR-B(3)*ZBAR
040790       REGSS=B(2)*SPXY+B(3)*SPZY
040800       REGMS=REGSS/2.
040810       RSSL=SSY-SPXY**2/SSX
040820       RSSQ=SSY-REGSS
040830       IRSSQ=KOUNT-3
040840       RMSQ=RSSQ/IRSSQ
040850       RSQ=REGSS/SSY
040860       RNUM=KOUNT-3
040870       RSQ=RSQ-(1./RNUM)*(1.-RSQ)
040880       FRAC=RSQ*100.
040890       DIFFSS=RSSL-RSSQ
040900       FQTERM=DIFFSS/RMSQ
040910       FREG=REGMS/RMSQ
040920 C
040930 C
040940 C-----WEIGHTS ARE FREQ./ANTILOG(.5*FITTED MS).
040950       DO 703 I=ISTART,N
040960           VMS(K,I)=0.
040970           NJ=N1(K,I)
040980           IF (NJ.EQ.1) Z(I)=X(I)
040990           IF (NJ.GT.0) THEN
041000               VMS(K,I)=B(1)+B(2)*X(I)+B(3)*Z(I)
041010               VMS(K,I)=0.5*VMS(K,I)
041020               W1(K,I)=NJ/EXP(VMS(K,I))
041030           ENDIF
041040  703  CONTINUE
041050       WRITE(8,601)
041060       WRITE(8,606)K
041070  606  FORMAT(1X,'FOR SAMPLE ',I3)
041080       WRITE(8,601)
041090       WRITE(8,603)FREG,IRSSQ,FQTERM,IRSSQ
041100  603  FORMAT(1X,'F TEST FOR PARABOLA AND FOR II DEGREE TERMS',/,1X,'FOR'
041110      >       ,' PARABOLA:',T24,'F= ',F10.3,3X,'D.F.= 2 AND ',I3,/,1X,
041120      >     'FOR II DEGREE TERMS:',T24,'F= ',F10.3,2X,'D.F.= 1 AND ',I3)
041130       WRITE(8,601)
041140       WRITE(8,604)FRAC
041150  604  FORMAT(1X,'PARABOLA EXPLAINS ',F5.2,' PERCENT OF THE VARIATION')
```

```
041160       WRITE(8,601)
041170       WRITE(8,605)(B(J),J=1,3)
041180  605  FORMAT(1X,'CONSTANTS OF THE PARABOLIC EQUATION ARE:'/1X, 'B(0) = '
041190      >       ,F10.4,2X,'B(1) = ',F10.4,2X,'B(2) = ',F10.4)
041200       RETURN
041210       END
041220 C
```

SUBROUTINE PLOT Developed by Janet Crance from an unknown source.
SUBROUTINES SMPY, GMPRD, LOC and MINV are part of the Scientific Subroutine Package of the I.B.M. Application Program number 360A-cCM03X, edition GH20-0205-4.